High-Throughput Screening of Tribological Properties of Monolayer Films using Molecular Dynamics and Machine Learning

Co D. Quach,[1,3] Justin B. Gilmer,[2,3] Daniel Pert,[1,3] Akanke Mason-Hogans[1,3], Christopher R. Iacovella,[1,3] Peter T. Cummings,[1,3] and Clare M^{c}Cabe[1,3,4,*]

[1]*Department of Chemical and Biomolecular Engineering,* [2]*Multiscale Modeling and Simulation Center,* [3]*Interdiscplinary Materials Science, and* [4]*Department of Chemistry, Vanderbilt University, Nashville, TN, 37235, USA*

* Author to whom corresponding should be addressed: c.mccabe@vanderbilt.edu

Abstract

Monolayer films have shown promise as a lubricating layer to reduce friction and wear of mechanical devices with separations on the nanoscale. These films have a vast design space with many tunable properties that can affect their tribological effectiveness. For example, terminal group chemistry, film composition, and backbone chemistry can all lead to films with significantly different tribological properties. This design space, however, is very difficult to explore without a combinatorial approach and an automatable, reproducible and extensible workflow to screen for promising candidate films. Using the Molecular Simulation Design Framework (MoSDeF), a combinatorial screening study was performed to explore 9747 unique monolayer films (116,964 total simulations) and a machine learning model using a random forest regressor, an ensemble learning technique, to explore the role of terminal group chemistry and its effect on tribological effectiveness. The most promising films were found to contain small terminal groups like cyano and ethylene. The machine learning model was subsequently applied to screen terminal group candidates identified from the ChEMBL small molecule library. Approximately 193,131 unique film candidates were screened, with approximately a 5 order of magnitude speed-up in analysis compared to simulation alone. The machine learning model was thus able to be used as a predictive tool to greatly speed up the initial screening of promising candidate films for future simulation

studies, suggesting that computational screening, in combination with machine learning, can greatly increase the throughput in combinatorial approaches to generate *in-silico* data and then train machine learning models in a controlled, self-consistent fashion.

Introduction

Monolayer films have shown promise as a means of reducing friction and wear of mechanical devices with nanoscale surface separations (*e.g.,* nano- and micro-electromechanical systems, NEMS and MEMS).[1,2] Such films are highly tunable through modification of their terminal group chemistries, backbone chain length, backbone chemistry, and film composition, all of which have been demonstrated to impact their tribological effectiveness along with other properties, such as durability, solvent interactions, and thermal response.[1–5] This tunability presents a rich chemical parameter space that can be explored for the optimization of film properties as well as gleaning useful information about the quantitative structure-property relationships (QSPR) of these systems, to develop better predictive models for design considerations.[6,7] Of these various modifications, the chemical and physical characteristics (descriptors) of the terminal groups plays a dominant role in the tribological response. For example, Yu *et al.*[4] showed that phenyl-terminated monolayer thin films yield higher frictional forces than methyl-terminated films, explained by the phenyl groupsô ability to twist (these can be thought of as structural molecular descriptors according to QSPR).[6] Hydroxylated and carboxylated thin films have been shown to have high frictional and adhesive forces relative to methyl-terminated films, attributed to their ability to form inter-monolayer hydrogen bonds during contact (physicochemical descriptors according to QSPR).[5,6] Similar trends found in molecular dynamics (MD) simulations further support these relationships.[8–10] Moreover, this interfacial region may feature not just a

single terminal group chemistry but instead multiple chemistries. For example, experiments by Brewer *et al*.[5] demonstrated that a methyl-functionalized microscope tip in contact with either hydroxyl or carboxyl terminated monolayers results in a lower coefficient of friction (COF) compared to the same tip in contact with a methyl terminated monolayer. Monolayers composed of two or more terminal group chemistries within the same layer, at varied relative compositions, may also provide a means to further tune and improve performance. For example, computational studies by Lewis *et al*.[11] for monolayers composed of methyl terminated alkanes mixed with perfluoroalkanes showed a regime where the COF was reduced compared to either pure component system. There are a multitude of complex relationships between these various chemical/molecular descriptors when translated to monolayer polymer systems as hinted at by Le *et al*.[6]

MD simulations are a useful tool to perform large-scale sweeps of the accessible parameter space to create an *in silico* self-consistent data set. Computational examination avoids the need to develop experimental synthesis techniques, which may be non-trivial and time intensive. Simulations can also more effectively reveal the intrinsic properties associated with defect-free films on pristine contaminant-free surfaces. This approach has been utilized to study and optimize various parameters describing the monolayer, such as backbone chain length and chain densities.[8] Our recent development of the Molecular Simulation and Design Framework (MoSDeF[12]) and the development of the Signac Framework[13,14] by Glotzer *et al.*, enables the large-scale screening of soft matter systems, allowing the reproducible initialization and parameterization of systems, and the management of large dataspaces. These tools have been used to perform large scale screening studies of soft matter systems in several recent papers[8,15] as well as used to fully capture the provenance of simulation workflows for increased reproducibility in other work.[16,17]

However, the vast parameter space to be explored for monolayer films would still make

brute force computational screening impractical. Instead, a promising approach is to combine computational screening with machine learning (ML) techniques in order to accelerate and better direct the exploration of the parameter landscape.[8] That is, use computational screening to gather sufficient data to train predictive ML models (thus minimizing the number of computationally expensive simulations), and subsequently using the ML models to predict the properties of new systems and guide the screening towards film chemistries with desirable properties. This approach has been utilized in other various studies with great success, such as developing predictive models that have errors lower than hybrid Density Functional Theory (DFT) methods[18,19], that learn and predict various protein folding events and structures[20–24], that use active learning to direct iterative optimizations and uncover optimal targets like structure[25–29] and to accelerate the discovery of novel monomers and polymers for favorable macroscopic properties[25,30–35]. Different ML models and techniques are applied but all demonstrate a useful approach to leverage *in silico* data from molecular simulations and experimental data (when available). As the power of *in silico* data is further realized for ML models and predictive design, being able to rapidly generate, screen, learn, and predict from these data will be powerful tools for computational and experimental researchers alike.

In prior work, we developed a screening framework to explore the role of terminal group chemistry on thin film tribological response under shear, enabling computational screening studies to be performed over a multitude of terminal group chemistries for contacting monolayers undergoing shear using non-equilibrium molecular dynamics (NEMD) simulations.[8] Uniform monolayers with 16 different terminal group chemistries were examined, with each monolayer terminal group chemistry independently varied, allowing key trends and several chemistry combinations that provided favorable tribological performance, *i.e.,* both low COF and low F_0, to

be identified. Furthermore, data from 100 different monolayer terminal group combinations were used to develop, train, and test ML models that allow COF and F_0 to be predicted with good accuracy solely from the chemistry of the terminal groups expressed as SMILES strings (discussed in detail in the Methods section). The success of these models, trained from a relatively small dataset, suggests that this approach can be used to prescreen computational space and accelerate the identification of films with favorable properties, assuming the dataspace is diverse enough to minimize the chance of overfitting. However, to determine if tribological performance could be further improved ï for example, by mixing terminal groups together within a film ï several key questions regarding the use of ML in a predictive capacity for monolayer films remain. Specifically, (1) what is the minimal data set required in order to adequately train ML models for tribological properties; (2) how transferrable are the ML models to systems with other chemistries and film compositions not included in the training data; and (3) can the models be used to prescreen the design space and if so, what level of accuracy can be achieved with a reasonable amount of training data?

To address these questions and further probe the tribological design space of thin films and the relationships between terminal group chemistry, thin film composition, and tribology, here we consider systems in which one monolayer consists of a single unique terminal group, and the other monolayer contains a mix of two unique terminal groups, allowing the relative composition of the two groups to be varied. For these designs, a pool of 19 different terminal groups were considered resulting in 9747 unique monolayer combinations, when considering the mixing ratio of terminal groups in the mixed monolayer, as will be discussed in detail in the Methods section and Fig. 1. In the Methods section we provide an overview of the computational approach, focusing on the simulation workflow, analysis methods, and the ML model. In the results section, we present the

data generated from the MD screening and identify key terminal groups and combinations associated with improved tribological performance. We then develop and evaluate the dependence of the ML models on the training set used, assessing the effectiveness of using a ML algorithm to predict properties. Finally, we utilize the ML model to demonstrate the feasibility of screening large data spaces (193,131 unique systems, created from 621 chemistries from the CheMBL library[36,37]), investigating suitable strategies for utilizing ML models to guide future work. All relevant information to reproducibly generate this data and workflow is readily available and adaptable for others to use, following the principle of TRUE (Transferable, Reproducible, Usable by others and Extensible) simulations described by Thompson *et al*.[16] See the supplementary information for more details.

Methods

Simulation Models and Workflow

In all cases, the simulated system consists of two opposing amorphous silica surfaces, each coated with an alkylsilane monolayer film. Specifically, each surface has dimensions of 5 nm x 5 nm, constructed using the procedure outlined by Summers *et al*.,[38] and available as an mBuild script provided in the Supplemental Repository.[39] The silica surfaces have an average surface roughness of 0.11 nm, closely approximating the more computationally intensive synthesis mimetic simulation approach by Black *et al*. [38,40] 100 alkylsilane chains are chemically bonded to each surface, with an in-plane surface density of 4 chains/nm^2; this surface density is consistent with prior computational studies[8,38,40] and experiments[41–43] that estimate chain surface densities to be between 4.0-5.0 chains/nm^2. Each alkylsilane chain is composed of a fully saturated 17 carbon backbone that is capped with a terminal group that can be easily varied computationally (see Fig. 1b). The remaining undercoordinated oxygens at the surface are changed to hydroxyl groups to

mimic surface oxidation. Of the two surfaces in the dual monolayer systems, the bottom surface is homogeneous (singular terminal group), while the top surface contains a mixture of two types of alkylsilane chains, differing by their terminal groups. The mixing ratios for the top monolayers considered in this study are 25:75 and 50:50. The pool of 19 different terminal group chemistries investigated are shown in Fig. 1b; this adds 3 additional terminal group chemistries to those considered by Summers *et al.*[8] The uniform bottom monolayer and the mixed top monolayer can be composed of any combination of groups from Fig. 1b, with the constraint that the two groups in the mixed monolayer must be different. In total 12,996 combinations ([19 terminal groups in uniform layer] * [19 * 18 terminal group combinations in mixed layer] * [2 composition ratios]) were considered; this translates to a total of 116,964 simulations (12,996 * 3 * 3) when factoring in the composition ratios studied, the 3 normal loads, and 3 replicates considered for each system. Of the 12,996 systems considered, 3249 systems with the mixing ratio in the top monolayer of 50:50 were duplicated during the screening and thus such combinations had 3 additional replicates; in total 9747 unique combinations (19 * 19 * 18 of 25:75 systems + ½ * 19 * 19 * 18 of 50:50 systems) were considered. We also note that a small subset of simulations (less than 1% of the total) failed to complete due to unstable initial configurations, but in all cases, each unique system composition reported includes at least 3 replicates.

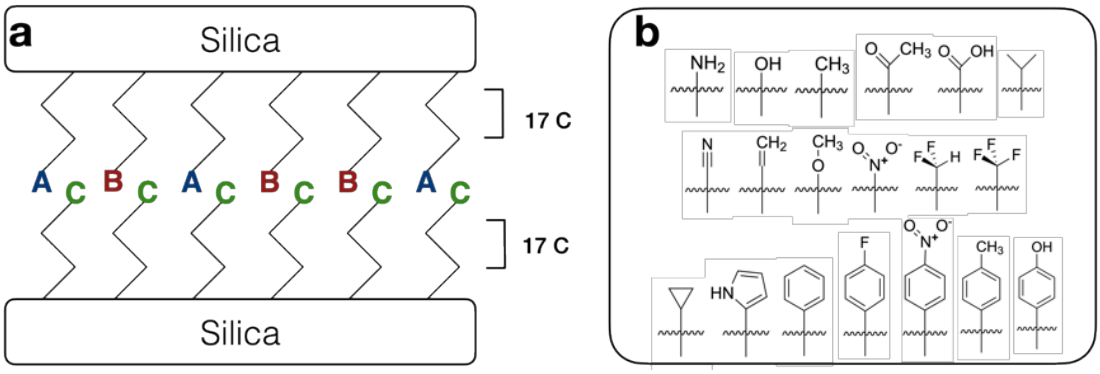

Figure 1. a) Simplified schematic of the systems studied. The top monolayer is a mixture of two types of terminal groups chemistries (A and B), studied at two different mixing ratios (25:75, 50:50), while the bottom monolayer is homogeneous (chemistry C). b) Depiction of the 19 different chemistries considered. From top to bottom, left to right, the terminal groups are amino, hydroxyl, methyl, acetyl, carboxyl, isopropyl, cyano, ethylene, methoxy, nitro, difluoromethyl, perfluoromethyl, cyclopropyl, pyrrole, phenyl, fluorophenyl, nitrophenyl, toluene, phenol.

Each monolayer system was prepared using the MoSDeF software suite[12,44,45] (see the Supplemental Information section for additional information). The initialization of the monolayer structure is encapsulated as an mBuild recipe,[44,46] which preserves the entire process used to construct the monolayer structure. The foyer library[45,47] was used to atom type and parameterize each system with the Optimized Potential for Liquid Simulation - All Atoms (OPLS-AA) forcefield.[48] Parameters for the alkylsilane chains were taken from GROMACS 5.1[49,50] and those for the silica surface from Lorenz *et al*.[51] The force field XML file used by foyer is provided in the Supplemental Information section and can be accessed from the Supplemental Repository.[39] The project workflow as a whole was managed using the Signac Framework.[14,52] The use of the MoSDeF framework in addition to the Signac Framework, ensures that all scripts and input parameters used to initialize the systems, submit the systems for simulation, and analyze the systems are captured and preserved, ensuring the simulations are TRUE (Transparent, Reproducible, Usable by Others, and Extensible).[16] All scripts and parameter files are available in the associated GitHub repository (see the Supplemental Information).[39]

MD simulations were performed using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) and GROMACS simulation engines.[49,50,53,54] LAMMPS was solely used to relieve the system of initial high-energy configurations and possible overlaps.[53,54] The more stable structure generated was then fed to GROMACS to perform the rest of the simulation workflow, starting with energy minimization following a steepest descent algorithm, followed by a 1 ns equilibration in the canonical (NVT) ensemble at 298K using the Nóse-Hoover thermostat.[49,50,55] An NVT simulation was then performed at 298 K in which the two surfaces were brought into contact by applying a constant normal force of 5 nN along the *z* direction to the bottom surface over 0.5 ns, allowing for the distance between the two surfaces to reach a steady state. After compression, shearing simulations (with surfaces moving at relative speed of 10 ms^{-1}) were performed at 3 different normal loads of 5 nN, 15 nN, and 25 nN. Specifically, the shearing process is simulated by pulling a ghost particle, which is coupled to the top surface via a harmonic spring with a spring constant of 10,000 kJ/(mol nm^2), in the *x* direction at 10 ms^{-1}. The shear is simulated for 10 ns, and the last 5 ns is used for analysis (production regime). The particle-particle particle-mesh (PPPM) algorithm was used to calculate the long-range electrostatic interactions, using a force and pressure correction in the *z*-dimension to support slab geometries; systems are periodic in the monolayer plane.[50,56]

Calculation of Tribological Properties

The coefficient of friction (COF), ‘, and adhesion force, ”Q were calculated from the last 5 ns of the simulation trajectory from each system under shear using the modified law of friction given by Equation 1,

”Q سي ”Q طى ‘ ”Qعي **(1)**

Here ″Q, ″Q$ ‘ and ″Q¡ represent the frictional force, the adhesive force, the coefficient of friction, and the normal force, respectively.[57] A linear regression of the average friction force (ordinate) versus normal load (abscissa) can be used to calculate the COF from the slope and ″Q$ from the intercept of the regression line with the ordinate axis. The friction force is calculated by summing all the forces in the direction of shear on one of the monolayers every 1 ps and averaged over the last 5 ns of the simulation.

Machine Learning Model

The dataset is analyzed using the random forest regressor as implemented in the scikit-learn library, consistent with our prior work in Summers *et al*.[8,58] Provided a training set, a set of input parameters and expected outputs, the random forest ensemble model will create a series of decision trees, each generated from a sub-sample of the training data set.[59,60] The trained random forest model then makes predictions by averaging the predicted values from the component decision trees. Each predictive model will rank the importance of each of the input parameters based on how a given input affects the final prediction. This ranking unveils information regarding properties that play an important role in predicting the tribological properties of the monolayer (feature importance), making this method advantageous for screening/discovery research. All of the random forest models in this study have 1000 trees, ensuring the predictions converge in a reasonable amount of time.[60] In the forest, each decision tree is allowed to expand until all leaves are pure (choosing splits that decrease impurity defined by the Gini impurity). All models used mean squared error (MAE) as error criterion during training. Each random forest model, and its subsequent decision trees, are trained with 32-42 features, which are molecular descriptors calculated through RDKit.[61] This setup is consistent with previous study by Summers *et al.*[8],

allowing for direct comparison between these studies, focusing on the accuracy of the models and feature importance ranking determined from the two sets of data. An effort to optimize the parameters resulted in insignificant improvement in model accuracy and thus the original values were retained for better comparability with prior work; further details can be found in the SI section (see Figs. S1-S3).

The chemical and physical input parameters for the ML model are supplied by the RDKit cheminformatics library.[61] The COF and F_0 calculated from the simulations are the expected outputs (*i.e.,* targeted properties) for the random forest ensemble to predict. The training procedure of these predictive models is adapted from that described in previous work.[8] Briefly, each of the systems in the training set can be represented by a set of SMILES strings,[62] describing the terminal group chemistry. Each terminal group is represented by two SMILES strings: one of a hydrogen capped structure and one of a methyl capped structure. The SMILES strings are used to calculate molecular descriptors that characterize the chemical and physical properties of chemical structures, via the RDKit cheminformatics library.[61] These descriptors fit into four categories: size (*e.g.,* molecular weight), shape (*e.g.,* inertial shape factor), complexity (*e.g.,* degree of branching), and charge distribution (*e.g.,* topological polar surface area). The SMILES string of the hydrogen-capped terminal group is used to calculate descriptors relating to shape, while the SMILES string of the methyl-capped terminal group is used to calculate the remaining descriptors. While shape characteristics can be sufficiently modeled with a hydrogen-capped structure, properties that involve charge distribution among others are better represented if they mimic the actual structure the terminal groups are attached to; a methyl terminus was found to be a sufficient approximation of the alkyl chain for these measurements.[8] Through this process, each chemical structure is represented by 53 descriptors, summarized in the Supplemental Information section (see Table S1).

The molecular descriptors for the top and bottom monolayers are first calculated independently. Descriptors for the top monolayer, with two terminal groups, are the weighted average (by relative composition) of its component terminal groups, while descriptors for the bottom monolayer are the molecular descriptors of its singular terminal group. This representation can have limitations when used to describe monolayers, since it does not encode information regarding connectivity of constituent chains and distribution pattern.[63] However, since we are mainly interested in the contribution of different terminal group chemistries, making up the inter-monolayer regions/interfaces, our method of calculating the molecular descriptor was found to be sufficient to encapsulate information for the region of interest, with an assumption that the two terminal groups in the top monolayer are randomly distributed. These are then combined, storing the mean and minimum of each descriptor of the system, totaling 106 descriptors ([53 metrics]*[2 corresponding to min and mean]), which has been demonstrated in previous work to encapsulate the most important features of these systems.[8] These "molecular fingerprints" later undergo a dimensionality reduction process to remove descriptors whose values have low variance and reduces highly correlated descriptors. Specifically, descriptors whose values are at least 90% correlated will be reduced to only one attribute (descriptors are sorted alphabetically), while descriptors whose variance is below 2% are also removed. This step reduced the number of descriptors (or effective fingerprint) of each system to be between 32 and 45, with a mean of 37 ± 3. The number of effective fingerprint descriptors decrease as the size of the training data increase, since some correlations may not manifest with a smaller data set. This dimensionality reduction process is consistent with that used in Summers *et al.*[8] using the source code hosted in a GitHub repo.[64] The reduced list of molecular descriptors are

then used as the input parameters to the ML models. The process of determining molecular ñfingerprintò of each system is summarized in

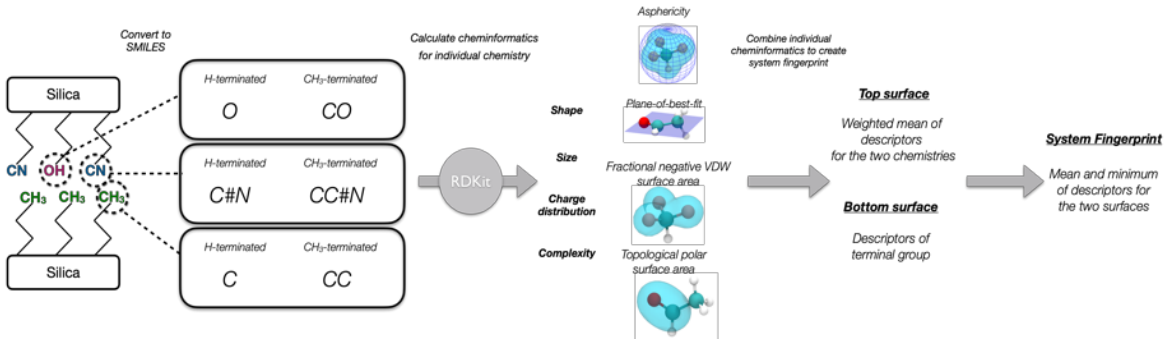

Figure 2. Process of generating the molecular descriptors (fingerprints) of the dual monolayer systems. Component terminal group chemistries of each top and bottom monolayer are represented by an H-terminated and methyl (CH_3)-terminated SMILES string, which can be used by RDKit to calculate corresponding molecular descriptors. For this study, we consider a total of 53 descriptors (listed in Table S1 is the SI), which can be grouped into 4 categories, namely, shape, size, charge distribution, and complexity. The weighted averages of these descriptors are then calculated to represent their corresponding surface, which in turn, will be used to determine the fingerprint of each system. Figure adapted from Summers *et al* [8].

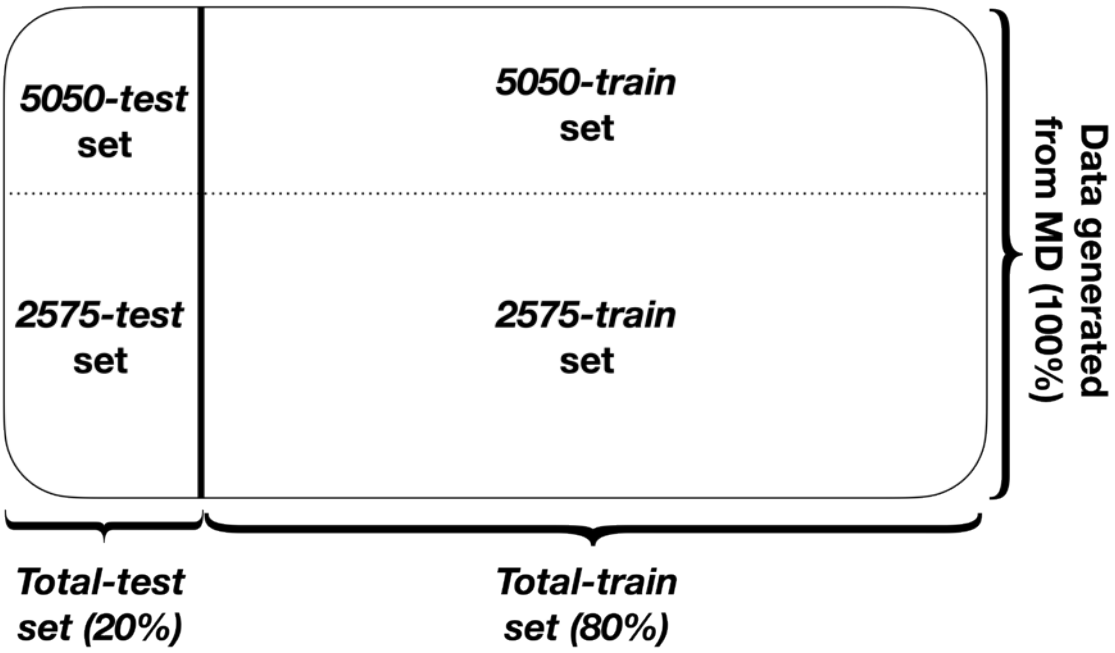

Figure 3. Summary of the data splitting process. The MD simulation data are split into two subsets: a *total-test* set (20% of the data set) and a *total-train* set (80% of the data set). The testing set is subdivided to create the *5050-test* and *2575-test* sets. The training set is also subdivided to create the *5050-train* and *2575-train* sets.

The total simulation data set is split into subsets as shown graphically in Fig. 3, as a means of examining different aspects of the training and transferability of ML models. In all cases, the data are split such that approximately 20% of the data are reserved for testing purposes, while the remaining 80% are used for training the model. When considering all composition ratios, the training and test sets are labeled *total-train* and *total-test*, respectively. We note that the most-favorable systems, defined as those with both low COF **and** F_0 values, of the entire simulation dataset (see Table 1) are included in the *total-test* set and removed from the training procedure. This allows us to evaluate the ML models ability to re-identify these systems from a test set, which will be explored in the body of this work. As shown in Fig. 3, the dataset can be further broken down by composition ratio, with the 50:50 and 25:75 mixing ratios considered separately, *e.g., 5050-test/5050-train* and *2575-test/2575-train*. To examine the role of training set size on the performance of the ML model the *total-train* and *5050-train* sets are further subdivided to create training sets with fewer data points. Since COF and F_0 of these films have been shown to have little correlation,[8] they are considered independently in the development of the ML models. To ensure that that the entire range of COF and F_0 are being properly sampled (Fig. 4), each training set is binned based upon the values of the COF or F_0 into 10 equal size quantiles. Data are then randomly selected from each bin to create training sets of varying size. For each training set size, 5 variations are created that differ only by the random seed used when sampling from the corresponding master training set, *e.g., total-train* and *5050-train*. Distributions of individual training sets are examined to ensure that they resemble the distribution of the full data set. Each training set is then used to train and create a predictive ML model that predicts either COF or F_0.

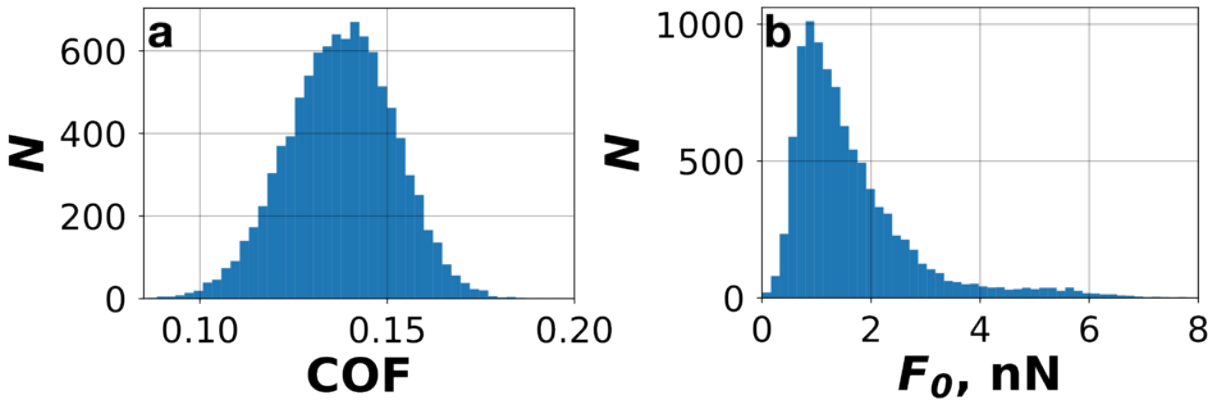

Figure 4. Distribution of a) COF and b) F_0 for systems considered in this study, obtained from MD simulations.

Results and Discussion

Considering first the results of the high throughput screening MD simulations, including those performed in the current study and in Summers *et al.*,[8] we identify 22 monolayer designs that provide favorable frictional properties, *e.g.,* those that have low simulated COF and F_0 values (see Table 1 and Fig. 5). This list was created by the intersection of the best 500 systems ranked from lowest to highest COF (values ranging from 0.074 to 0.114) with the best 500 systems ranked from lowest to highest F_0 (values ranging from 0.007 nN to 0.541 nN). We first note that, in general, these designs are in agreement with the conclusion obtained by Summers *et al*. from a study of a considerably smaller dataset where it was noted that the COF of monolayers is primarily affected by the shape and size of the terminal group, with chemistries of small sizes and simple shapes (*e.g., sp* hybridization) exhibiting the lowest COF.[8] Summers *et al*. also noted that the F_0 is most strongly affected by charge distribution, with polarity and hydrogen bonding both increasing the F_0.[8] In agreement with these findings, we observe that a majority of the systems identified (19 out of 22) consists of a cyano homogeneous monolayer. The cyano group is small in size, has *sp* hybridization

and does not readily form hydrogen bonds, characteristics that fit with previous work to identify chemistries that can lower the COF and F_0 of monolayers. We also note most systems in Table 1 are made up of 3 components and only one system that consists of two homogeneous monolayers (System 1 in Table 1), which was simulated in the Summers *et al.* work.[8] This result suggests a slight advantage to having mixed monolayer designs. However, we also recognize that the data set is dominated with mixed monolayers compared to homogeneous monolayers, therefore the best performing systems are likely the result of the much larger representation of mixed monolayer systems compared to the homogeneous systems. Nonetheless, mixed monolayer systems could provide extra flexibility during the design process and allow for the optimization of other properties, such as thermal stability or environmental interactions, depending on the specific application, giving these designs advantages over homogeneous monolayers.

Table 1. 22 most-favorable systems determined by the intersection of the top 500 systems ranked by their COF and the top 500 systems ranked by their F_0. The COF and F_0 mean values and standard deviation (std) are calculated from the 3 replicates.

	Terminal Group A	Terminal Group B	Terminal Group C	A Fraction	B Fraction	COF - mean	COF - std	F_0 , nN - mean	F_0 , nN std
1	cyano	cyano	isopropyl	0.5	0.5	0.1032	0.0063	0.4768	0.1075
2	cyclopropyl	ethylene	cyano	0.5	0.5	0.1086	0.0142	0.4498	0.2820
3	difluoromethyl	isopropyl	cyano	0.5	0.5	0.1100	0.0098	0.5292	0.4261
4	difluoromethyl	methyl	cyano	0.5	0.5	0.1128	0.0036	0.4640	0.1585
5	ethylene	isopropyl	cyano	0.5	0.5	0.1109	0.0108	0.2161	0.3768
6	ethylene	perfluoromethyl	cyano	0.5	0.5	0.1093	0.0119	0.3297	0.3124
7	isopropyl	methoxy	cyano	0.5	0.5	0.1113	0.0144	0.5313	0.5019
8	isopropyl	methyl	cyano	0.5	0.5	0.1121	0.0182	0.4041	0.3252
9	isopropyl	perfluoromethyl	cyano	0.5	0.5	0.1030	0.0181	0.2978	0.1960
10	difluoromethyl	carboxyl	isopropyl	0.25	0.75	0.1117	0.0021	0.1944	0.8319
11	difluoromethyl	isopropyl	carboxyl	0.25	0.75	0.1105	0.0078	0.5375	0.7366
12	difluoromethyl	methyl	cyano	0.25	0.75	0.1126	0.0041	0.4967	0.0876
13	ethylene	isopropyl	cyano	0.25	0.75	0.1046	0.0184	0.5122	0.1374
14	isopropyl	cyclopropyl	cyano	0.25	0.75	0.0898	0.0119	0.4354	0.1522
15	isopropyl	perfluoromethyl	cyano	0.25	0.75	0.1077	0.0073	0.3868	0.1580
16	methoxy	cyano	ethylene	0.25	0.75	0.1086	0.0053	0.4346	0.4340
17	methyl	isopropyl	cyano	0.25	0.75	0.0942	0.0148	0.4181	0.2017
18	perfluoromethyl	cyclopropyl	cyano	0.25	0.75	0.1138	0.0070	0.3104	0.1799
19	perfluoromethyl	ethylene	cyano	0.25	0.75	0.1067	0.0002	0.5315	0.1965
20	perfluoromethyl	isopropyl	cyano	0.25	0.75	0.0947	0.0114	0.5366	0.2037
21	phenyl	isopropyl	cyano	0.25	0.75	0.1098	0.0156	0.4825	0.3196
22	toluene	ethylene	cyano	0.25	0.75	0.1119	0.0134	0.4907	0.6589

[A] Coefficient of friction
[9] Adhesion force

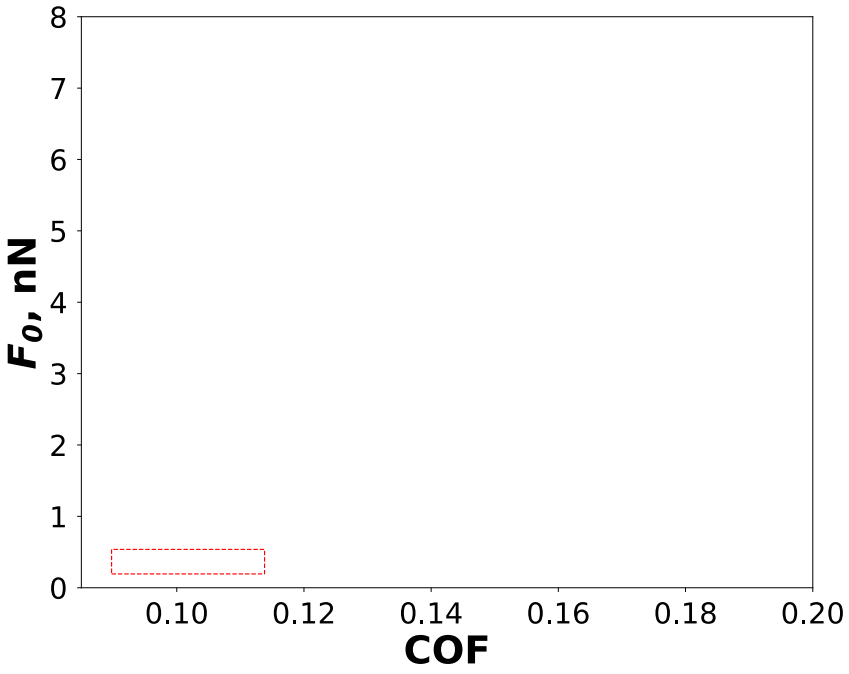

Figure 5. Distribution of simulated systems based on their COF and F_0 values. The 22 most-favorable systems, listed in Table 1, corresponds to data points confined within the red dashed box in the lower left quadrant of the figure.

Using the simulation data, we now explore combining ML techniques with MD simulations to perform high-throughput screening of monolayer systems. In Summers *et al*., the random forest regressor algorithm was applied to create predictive ML models to estimate the frictional properties of alkylsilane monolayers capped with different terminal group chemistries.[8] The ML models were trained on simulation data for homogeneous monolayers with 16 distinct terminal groups, resulting in a relatively small data set, containing only 100 data points. The models were then applied to a test set and compared to the COF and F_0 results obtained for the same systems directly from MD simulation to determine the accuracy of the ML models. This comparison can be readily visualized by plotting the tribological properties obtained from the ML models against the values calculated from the MD simulations; the coefficient of determination (R^2) and the mean absolute percentage error (MAPE) of the plots are used to quantitatively measure the accuracy of the ML predictions. The R^2 is commonly used/reported to quantify the correlation between the

simulated and predicted values, and MAPE provides error metrics that scale by the prediction values.[65] Using the additional simulation data generated herein, we can now better assess the feasibility of using ML to predict tribological properties and determine the amount of data necessary to create models that can make sufficiently precise estimations, as described below.

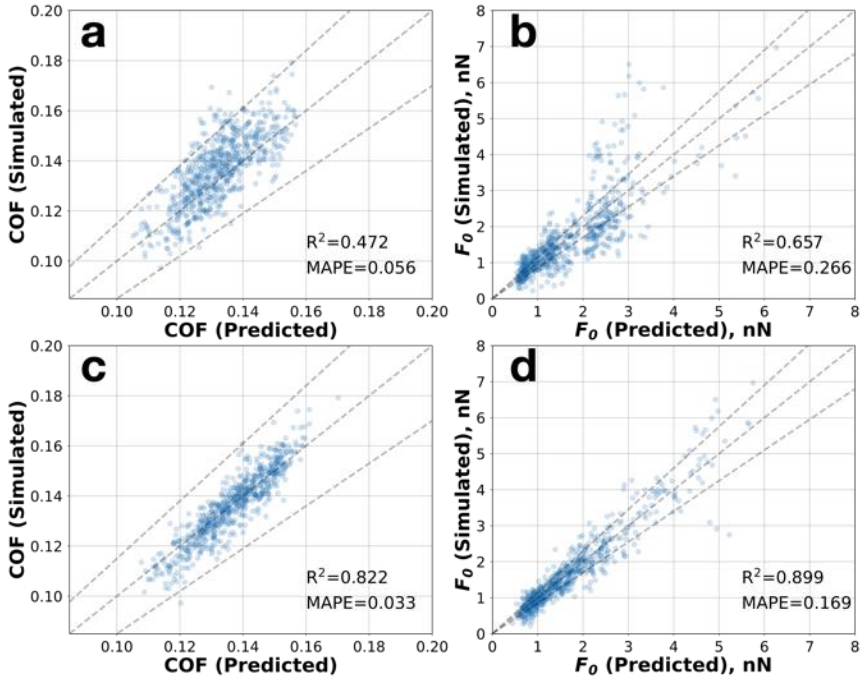

Figure 6. Predicted-versus-simulated plots for COF and F_0 for models trained with 100 simulation data points for uniform monolayers from Summers *et al.* [8] data set (a and b) and trained with 1000 data points randomly chosen from the *5050-train* set (c and d). The dotted line in the middle represents perfect prediction (y = x). The outer two lines represents the 15% variation from a perfect prediction (y = 1.15x and y = 0.85x). The coefficient of determination (R^2) and mean absolute percentage error (MAPE) are included. For each system (data point), the predicted properties are averaged from the 5 predictions made by the 5 ML models replicate, and the simulated properties are averaged from at least 3 simulations replicate.

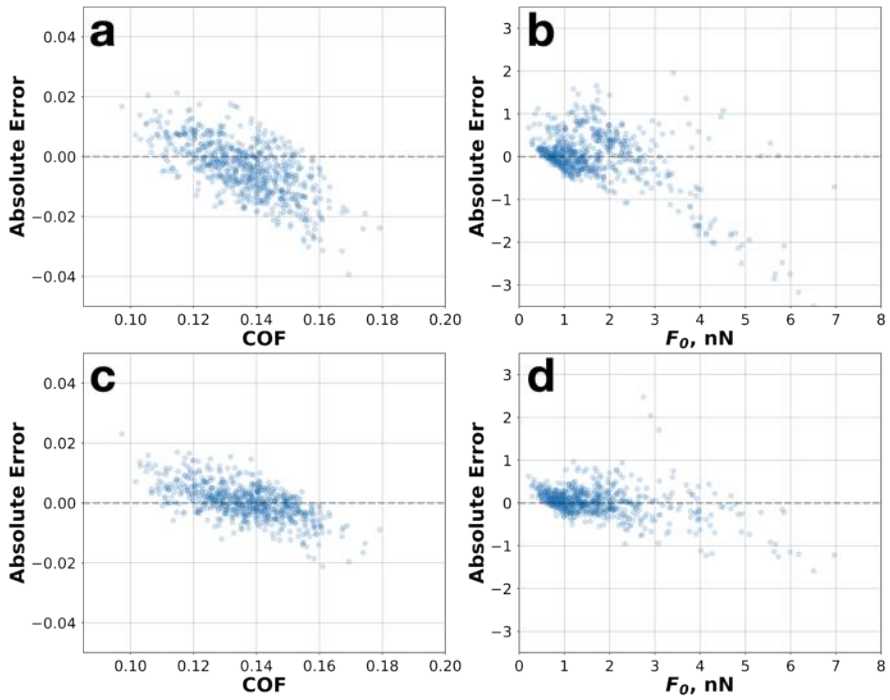

Figure 7. Deviations in predictions of COF and F_0 from ML models trained with 100 simulation data points for uniform monolayers from Summers *et al.* [8] (a and b) and trained with 1000 simulation data points randomly chosen from the *5050-train* set (c and d).

We first train a new set of ML models using 1000 data points from the *5050-train* set. The models are then applied to the *5050-test* set and, as described in the Methods section, compared to the COF and F_0 results obtained directly from the MD simulations in order to determine the accuracy of the ML models (see Fig. 6). Results for the ML models trained with the Summers *et al.*[8] data set applied to the *5050-test* set are also shown for comparison. When applied to the same testing set, the Summers *et al.* models provide R^2 values of 0.472 and 0.657 for COF and F_0 respectively, compared to 0.822 and 0.899 for COF and F_0 from the *5050-train* set. While the R^2 values are lower for the Summers *et al.* ML models, it is worth noting that the training data set did not include any information regarding mixed monolayer compositions; as such, the Summers *et al.* ML models still demonstrate impressive efficacy. This point is further demonstrated by their MAPE, where the Summers *et al.* models could predict COF of system with 0.056 (5.6%)

error and predict F_0 with 0.266 (26.6%) error. These MAPE values are higher but are still comparable with those produced by the new set of models, trained with 10-fold amount of data. Clearly our prediction of F_0 is less accurate in the higher adhesion regime than the lower adhesion regime, as seen in Fig. 6b and 6d, as was also observed in the prior work of Summers *et al.*[8] This is likely related to the challenges associated with capturing the ability of systems to form hydrogen bonds between contacting layers, although, since our primary goal in this work is to identify systems with low adhesion values, quantitative agreement in the higher adhesion regime is not required, as previously discussed in Summers *et al.*[8] Nonetheless, this result suggests that ML models trained with limited data could still provide meaningful estimation, and that the use of the random forest regressor may lead to models that are predictive for chemistries and compositions outside of the training set. It should be noted, however, that this relationship could be solely related to these monolayer systems and specifically to non-equilibrium shearing and may not be applicable to other non-equilibrium studies. The prediction deviation plots for these models are shown in Fig. 7. In general, we see that for lower values of COF or F_0, both models deviate slightly in the positive direction, meaning they predict a slightly higher value compared to simulation; as the value of either COF or F_0 increases, a negative deviation is observed with the ML models predicting slightly better performance than is observed in the MD simulations (see Fig. 7a, b). This skew in the predictions suggests that for favorable tribological conditions (*i.e.,* low COF and low F_0), the model will tend to overestimate the values, thus reducing the likelihood of incorrectly identifying poor performing films as viable options. This trend appears to be correlated to the size and distribution of the training set provided, with the trend becoming less apparent as the size of the training data set is increased (see Fig. 7c, d). Given that this behavior of the model minimizes the chances of exaggerating the performance of high performing systems (*i.e.,* those with low COF

and F_0), this suggests the predictive ML models can be more confidently used to screen over potential film candidates for possible applications. We also note that while the R^2 values for COF models are substantially smaller than those of F_0 models, which might suggest the latter models outperform their COF counterparts, their MAPE values indicate the opposite, where the F_0 models exhibit significantly greater percentage errors. This disparity could be attributed to the difference in the range of these two properties; while COF values span a small range of values from roughly 0.085 to 0.2, F_0 can take values from ~ 0 nN to 8 nN (see Fig. 4), which may affect how these metrics are calculated. Hence, it is important to recognize that that neither R^2 or MAPE values can directly relate to the predictive ability of the COF and F_0 models, though they can still be used to compare the performance of ML models of a similar type. Comparison of the feature importance of the two models at this point can be misleading, since the two set of models are trained with feature vectors of various size and components, as discussed in the Methods section. An extensive comparison feature importance of different models in this study is further discussed in the Supplemental Section (see Figs. S4-S7).

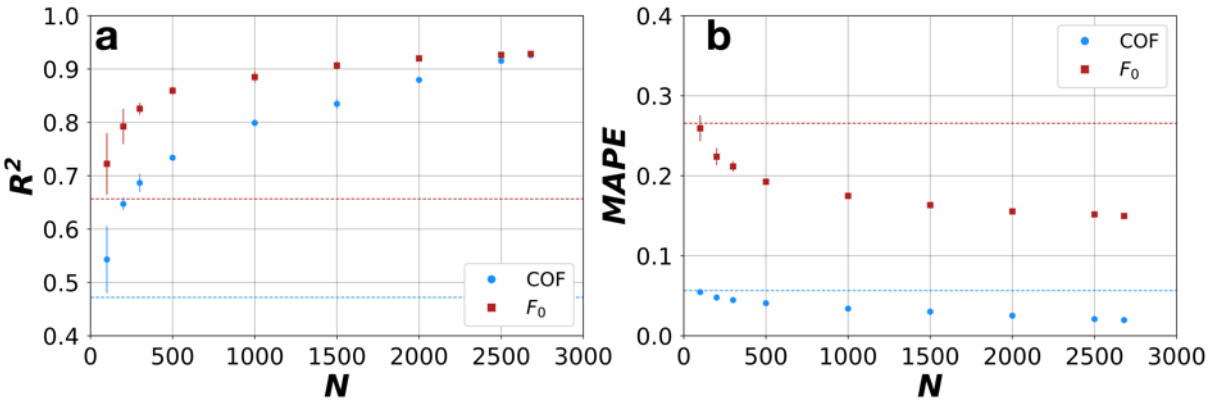

Figure 8. Correlation between the amount of data in the training data set (N) and the predictive ability of the models trained using the *5050-train* sets when applied to the *5050-test* set to predict the COF (blue, circle) and F_0 (red, square) quantified by R^2 (a) and MAPE (b). The dashed, horizontal lines, colored to correspond to its respective property in the key, show the predictive ability of the models trained using the Summers *et al.* [8] data set. For each data point, the metric (R^2/MAPE) are averaged from metric of individual ML models (5 replicates) when applied to the test set. The error bar of each point represents the standard deviation of the 5 ML models, each trained with different combinations of data.

We further examine the quality of the ML model estimations as a function of training set size by gradually increasing the amount of data used to train the ML models. We start with only using data from the *5050-train* set; by doing so, we can later evaluate the transferability of the model to the 25:75 systems, *i.e.,* can these predictive models provide similarly precise estimation of the frictional properties of systems with different designs. The R^2 and MAPE values for models trained on data sets of increasing size from the *5050-train* set are reported in Fig. 8. The results for each data set size are calculated from 5 models, each differing by the random seeds used when sampling from the *5050-train* set, and the standard deviation of the predictions of the 5 models are represented as error bars. The R^2 and MAPE of estimations made by the Summers *et al.* models are also shown in dotted lines for reference. From Fig. 8a, we can see that the accuracy of the ML model predictions improves rapidly as the training set size is increased, with the ML predictions roughly plateauing between 1000 and 1500 data points for which the R^2 values for COF are 0.799 ± 0.007 and 0.835 ± 0.011 and for F_0 are 0.885 ± 0.001 and 0.907 ± 0.007, respectively. Similarly,

in Fig. 8b, the MAPE of both COF and F_0 models also decrease gradually from 0.0546 ± 0.003 (5.46% ± 0.3%) for COF and 0.259 ± 0.016 (25.9% ± 1.6%) for F_0 at 100 data points, and level-off near 1000 data points, at 0.0339 ± 0.001 (3.39% ± 0.01%) for COF and 0.175 ± 0.003 (17.5% ± 0.3%) for F_0. While the predictive ability of the models does increase with the size of the training set beyond 1000 datapoints, the gains in accuracy are much less significant, suggesting one could achieve sufficiently accurate ML models even with a modest amount of data.

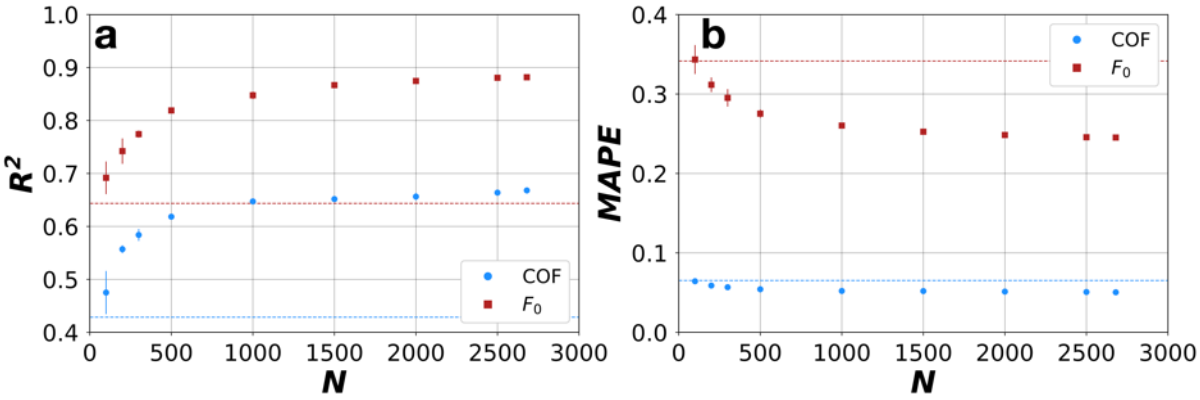

Figure 9. Correlation between the amount of data in the training data set (N) and the predictive ability of the models trained using the *5050-train* sets when applied to the *2575-test* set to predict the COF (blue, circle) and F_0 (red, square) quantified by R^2 (a) and MAPE (b). The dashed, horizontal lines, colored to correspond to its respective property in the key, show the predictive ability of the models trained using the Summers *et al.* [8] data set. For each data point, the metric (R^2/MAPE) are averaged from metric of individual ML models (5 replicates) when applied to the test set. The error bar of each point represents the standard deviation of the 5 ML models, each trained with different combinations of data.

We now examine the transferability of the ML algorithms in terms of their ability to predict the frictional properties of systems with different designs, *i.e.,* systems with a different mixing ratio on the top monolayer in the testing set than in the training set. Results from applying the ML models described above, trained solely with data from the *5050-train* set and the Summers *et al.* data set, on the *2575-test* set are reported in Fig. 9. From Fig. 9a, we observe that the R^2 values for COF and F_0 increase rapidly before plateauing, again at a training set size of approximately 1000 points, with values of 0.647 ± 0.001 and 0.848 ± 0.007 for COF and F_0, respectively. Similar trends are observed in Fig. 9b, where the MAPE of both COF and F_0 models rapidly decrease up until

1000 data points before leveling-off, with an error of 0.0521 ± 0.0 (5.21% ± 0.0%) for COF and 0.26 ± 0.003 (26.0% ± 0.3%) for F_0. While the accuracy is lower, *i.e.,* lower R^2 and higher MAPE, than that observed for the *5050-test* set (see Fig. 8), the agreement is promising, considering the ML models were not trained with data at these composition ratios. The plateau of the accuracy of the models is important, as it shows that improvements in accuracy of the models with larger data sets (as seen in Fig. 9) do not necessarily manifest themselves when the model is transferred to compositions outside of the original training set. Moreover, from Figs. 8 and 9, we notice the *5050-train* models trained with 100 data points also exhibit slightly better accuracy than the model trained with Summers *et al.*[8] data set, likely due to the inclusion of mixed-monolayer systems in their training sets. We note, for random forest regressors, the variety of data is more important in determining the quality of the predictions compared to the amount of data beyond a certain point, which is dependent on the complexity of the systems of interest; this point will be further examined for our specific systems in the Supplemental Information (see Fig. S8-S10). That is, there may be limited utility of using large training sets when trying to develop ML models to prescreen systems outside of the design space of the original training set. However, generating a set of systems with well distributed properties can be challenging and hard to estimate *a priori,* and hence may require more thoughtful design of the initial screening space as well as a more active learning approach to direct the screening space as the initial data is used to train the models.

The results in Fig. 9 suggest that the ML models are likely effective in predicting frictional properties of systems with design variations, *i.e.,* despite the noticeable decline in performance the models could likely be used as a high-level screen to sieve the parameter space. To further examine the capability of ML models in shortening the list of potential candidates to be simulated/synthesized, we examine the ability of the models to identify systems that exhibit

favorable tribological performance (*i.e.,* low COF or F_0). To quantify the ability of the model to predict favorable solutions, we calculate the intersection of the top performing systems predicted by the ML model and those via simulation; an ideally performing ML model would be able to identify all of, or a majority of, the best performing systems determined through simulation The ability of the ML model to accurately predict the systems with the favorable properties can be considered to be proportional to the percentage of the overlapping systems compiled from MD simulation and ML prediction. An overlap of 100% indicates complete agreement between the two methods, *i.e.,* ML and MD, while a low overlap value indicates lower agreement, and by extension, poorer predictive ability of the ML models. We note that this metric describes the ability of the ML model to accurately capture relative differences between systems of interest and does not necessarily require quantitative agreement between the ML model and corresponding MD simulations.

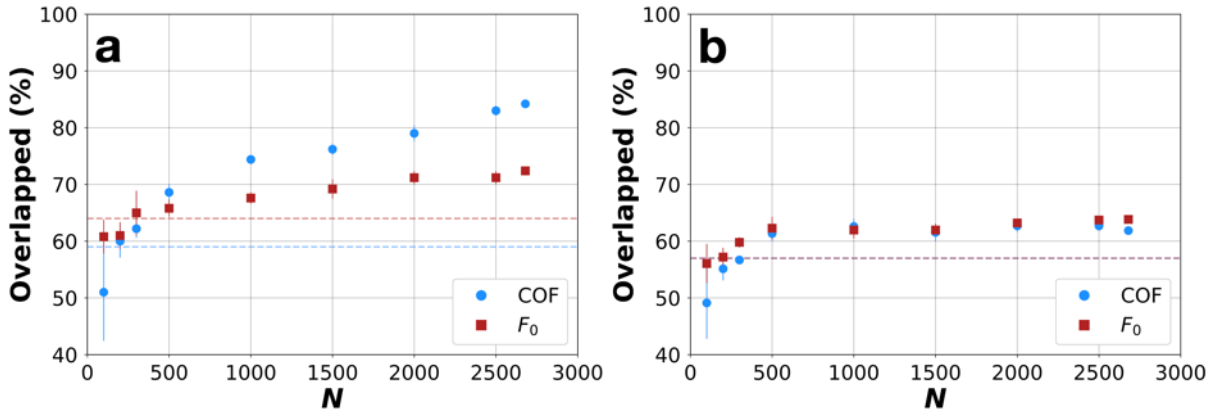

Figure 10. Intersection between the top 15% performing systems predicted by ML models, at various training set size, and top 15% performing systems calculated by MD simulations (*in silico* data) of the *5050-test* set (a) and the *2575-test* set (b). The systems are ranked by COF (blue, circle) or F_0 (red, square). The dashed, horizontal lines, colored to correspond with its respective property, show the predictive ability of the models trained using the Summers *et al.* [8] data set.

First, we first consider the ability of *5050-train* models in determining the best performing systems in the *5050-test* set in Fig. 10a. Systems in the set are first sorted separately, by the

numerical value of their COF and F_0 calculated from the simulations; the top 15% of these systems (*i.e.,* systems with the lowest COF or F_0 values) of each set are considered, corresponding to 100 chemistries for *5050-test* and 193 for 2575-test. These lists are then compared to the top 15% of systems predicted by the ML models, as a function of training set size, and the overlapping percentages are calculated. For COF, as the training set size of the model increases, so too does the fraction of top performing solutions predicted, achieving 74.4% ± 0.8% accuracy for models trained using 1000 data points and 83.0 ± 0.9% accuracy for models trained with 2500 systems; adhesion shows a weaker dependence on training set size, reaching 67.6% ± 1.1% at 1000 data points and 71.2% ± 1.2% at 2500 data points (see Fig. 10a). Putting these results into perspective, if we had utilized the *5050-train* model of 1000 data points to predict systems with the best performing properties from the *5050-test* set and only simulated those in the top 15%, we would have reduced the total number of additional screening simulations by 85%, while still identifying 74.4% of the best performing systems as ranked by COF, or 83% of the best performing systems ranked by F_0. On the other hand, if we reduced the number of systems to be simulated at random, we would only expect to detect 15% of the best performing systems, ranked by COF or F_0. In other words, this approach can significantly increase the odds of finding systems with the top performing tribological properties. We conduct the same analysis of the ability of the ML models to identify the best performing systems in the *2575-test* set, *i.e.,* focusing on the transferability of the models, and show the results in Fig. 10b. We observe predictions with an accuracy of roughly 61% for both the best performing systems ranked by COF or by F_0, almost independent of training set size (see Fig. 10b). Even though these accuracies are not as high as the models used on the *5050-test* set, they are still considerably higher than the 15% accuracy we would have expected if selecting systems at random. This suggests reasonable efficacy of using this approach to prescreen design

space. Even for models trained with limited amounts of data, *i.e.,* models trained with 500 data points whose accuracies are 61.3% ± 1.3% for COF models and 62.3% ± 2.0% for F_0, the predictions made should still be useful enough for prescreening and provide focused guidance to perform the next round of simulations, while reducing the computational costs substantially.

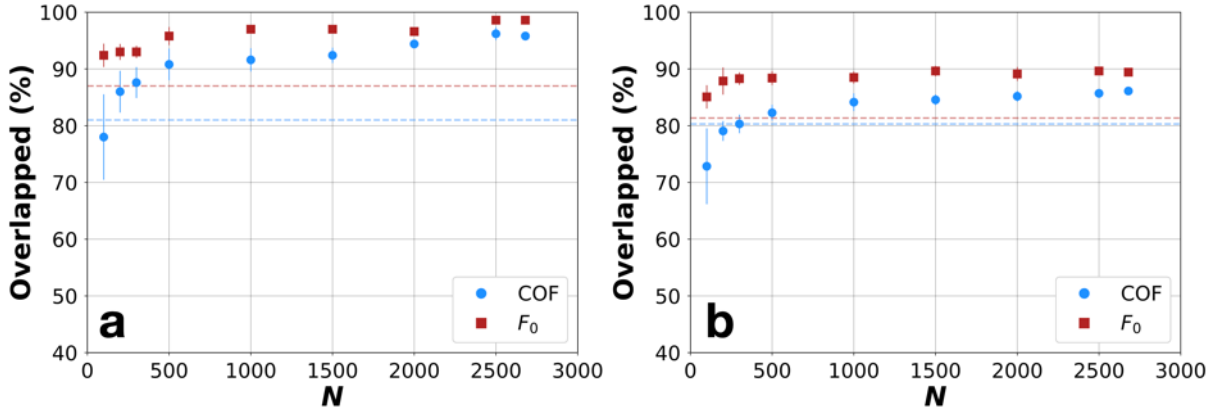

Figure 11. Intersection between the top 15% performing systems predicted by ML models, at various training set size, and top 30% performing systems calculated by MD simulations (*in silico* data) of the *5050-test* set (a) and the *2575-test* set (b). The systems are ranked by COF (blue, circle) or F_0 (red, square). The dashed, horizontal lines, colored to correspond with its respective property, show the predictive ability of the models trained using the Summers *et al.* [8] data set.

In Fig. 10 we have compared an equipercentile of the top performers determined through different techniques, *i.e.,* MD and ML. This approach, however, can potentially result in missing out in potential candidates, *e.g.,* a model that has an accuracy of 60% in determining the true best 15% systems will miss potentially 40% of the candidates. In practice, considering a larger list of top performing systems proposed by the ML models is likely to increase the number of top performers identified in the given parameter space; this is especially relevant for a quantity such as COF, where the overall numerical range is relatively small, *e.g.,* for *5050-test* the top 15% of systems range from 0.0972 ± 0.016 to 0.121 ± 0.010 and the top 30% only increases the upper bound very modestly to 0.129 ± 0.008 especially in context of the accuracy of the ML previously discussed; for F_0 the range for the top 15% is 0.216 ± 0.377 nN to 0.779 ± 0.135 nN, with the

upper bound increase to 1.002 ± 0.171 for the top 30%. For *2575-test* the values are similar where the top 15% for COF ranges from 0.090 ± 0.012 to 0.122 ± 0.014, with the upper bound increasing to 0.130 ± 0.011 for the top 30%, and for F_0, 0.085 ± 0.462 nN to 0.706 ± 0.410 for 15%, with the upper bound increasing to 0.951 ± 0.436 when considering the top 30%. Fig 11 plots the overlap between the top 15% systems (MD data), ranked by their simulated tribological properties and the top 30% performing systems predicted by the *5050-train* models. This set up demonstrates a significant increase in accuracy in predicting the best performing systems in both test sets. Specifically, in Fig. 11a, the accuracy of predicting top performing systems starts at 78.0% ± 7.5% and 92.4% ± 2.1% (for models trained with only 100 data points) to 91.6% ± 2.1% and 97.0% ± 0.6% (for models trained with 1000 data points) when predicting top systems ranked by COF and F_0, respectively. In Fig. 11b, the overlap percentages start at 72.8% ± 6.7% and 85.1% ± 2.1% (for models trained with 100 data points) to 84.1% ± 1.6% and 88.5% ± 1.1% (for models trained with 1000 data points). In other words, by considering the top 30%, the ML models can reduce the number of additional systems to be considered by 70%, while being able to accurately predict the bulk of the best performing systems, regardless of the mixing ratio and even for very small training set sizes. These results also present other criteria to consider in terms model accuracy and computational cost in terms of using ML models to prescreen a dataspace, as these results show that it may be more efficient to train a model with fewer datapoints but consider a larger range of predictions from the ML model (*e.g.,* simulating the top 30% predicted by the ML model). These results suggest a general approach to combine ML techniques with MD simulations, namely simulating a small set of systems, *e.g.*, about 5-10% of systems in the design space, using the simulation results to train predictive ML models, and then utilizing the ML models to screen over a wider range of potential systems, determining systems worthy of further investigation. Such an

approach could drastically minimize the number of total simulations needed, decrease the throughput time to scan the parameter space, enabling higher quality candidates to be screened much faster.

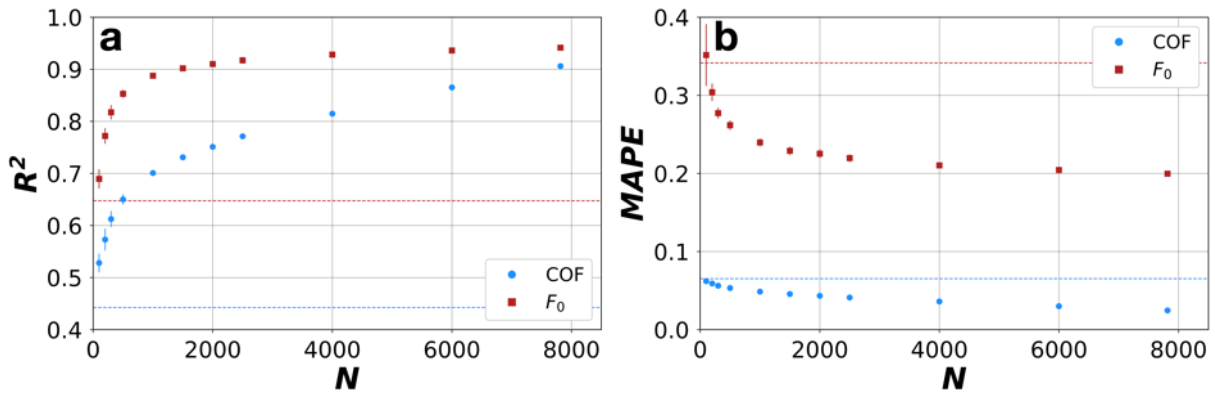

Figure 12. Correlation between the amount of data in the training data set (N) and the predictive ability of the models trained using the *total-train* sets when applied to the *total-test* set to predict the COF (blue, circle) and F_0 (red, square) quantified by R^2 (a) and MAPE (b). The dashed, horizontal lines, colored to correspond to its respective property in the key, show the predictive ability of the models trained using the Summers *et al.* [8] data set. For each data point, the metric (R^2/MAPE) are averaged from metric of individual ML models (5 replicates) when applied to the test set. The error bar of each point represents the standard deviation of the 5 ML models, each trained with different combinations of data.

Thus far we have considered models trained only using the *5050-train* data. One would assume that accuracy could be improved by training the models on a data set that also included those systems in the *2575-train* data set (*i.e.,* using the *total-train* set). Fig. 12 plots the scaling for R^2 (Fig. 12a) and MAPE (Fig. 12b) as a function of training set size when sampled from the *total-train* set and applied to the *total-test* set. As seen earlier, the rapid increase in accuracy occurs as the training set size is increased to 1000 data points; the improvements in adhesion are relatively minimal beyond that point, although considerable gains are observed for COF, with both measures attaining an R^2 value of >0.9 for a training set of size 7816 (see Fig. 12a). We also observe similar trends for MAPE in Fig. 12b, where the error quickly drops from 0.0593 ± 0.001 (5.93% ± 0.1%), for COF predictions, and 0.324 ± 0.033 (32.4% ± 3.3%), for F_0 predictions, at 100 data points to

0.023 ± 0.0 (2.30% ± 0.0%) for COF predictions, and 0.179 ± 0.0 (17.9% ± 0.0%) for F_0 predictions, at the maximum number of training data (7816). Focusing on the models trained with 1000 data points, predictions from the *total-train* set ML models applied to the *total-test* set achieve R^2 values of 0.701 ± 0.005 for COF and 0.888 ± 0.005 for F_0, which are slightly lower than the values obtained for the models trained with the *5050-train* set when applied on the *5050-test* set (see Fig. 8). This may appear to indicate that this set of ML models require more data to attain a similar level of predictive ability, especially for COF models. However, it is worth noting that these evaluations are done on two test sets of differing size and composition. Moreover, the *5050-test* set is a strict subset of the *total-test* set, so the latter includes a wider range of systems, and hence is deemed more challenging for the ML models. Thus, the relative performance of these models could not be directly compared at this point.

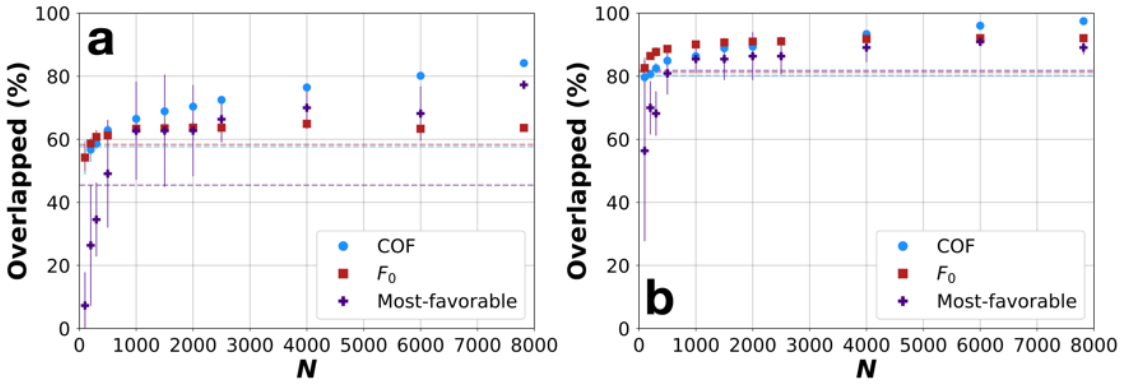

Figure 13. Intersection between the (a) top 15% performing systems, ranked by COF (blue, circle) or F_0 (red, square) or combined (purple, cross), of the *total-test* set determined from MD simulations and predicted by ML models trained with the *total-train* sets. The overlapped fraction of most-favorable systems is determined by comparing those predicted by ML models against those listed in Table 1. The error bar of each point represents the standard deviation of the 5 ML models, each trained with different combination of data. The dashed, horizontal lines, colored to correspond with its respective property, show the predictive ability of the models trained using the Summers *et al.* [8] data set.

To evaluate the accuracy of using the *total-train* models for prescreening parameter space, *i.e.,* their ability to determine best performing systems, we again calculate the agreement between the top 15% performing systems (398 total systems) from the *total-test* set as determined by the

MD simulations and the top 15% and 30% top performing systems predicted by our ML models, plotted in Fig. 13a and 13b. In Fig. 13a, we see a steady improvement in the accuracy of the COF predictions as nearly 8000 data points in the training set are used, achieving an overlapping of 84.2% ± 0.3%; F_0, exhibits a similar trend observed previously in Fig. 9b, with little dependence on training set size beyond 1000 points, maintaining an overlapping value of approximately 63%. Hence, increasing the number of data points to train the ML models can have a positive effect on the predictive ability for some properties, but the large amount of training data needed to improve accuracy may ultimately negate potential performance gains in terms of screening. Recall that for a training set size of 1000 data points, the *5050-train* model could predict >65% of high-performing 50:50 systems and >60% of high-performing 25:75 systems, ranked by either COF or F_0 when conducting similar analyses (see Fig. 10). In Fig 13b, we perform similar overlap analyses with an extended list of top performing systems predicted by ML models (top 30%). When determining top COF/F_0 systems, we observe that the overlapping fraction rapidly increases to above 80% at N=500, and subsequently plateaus, displaying minimal increase in their accuracy past this point, similar to what was previous observed in Fig. 10.

In addition, we introduce a new category, most-favorable systems, defined as those with both low COF **and** F_0. This list of favorable systems is generated from the intersection of the top 15% (Fig. 13a) or 30% (Fig. 13b) systems ranked by their predicted COF **and** F_0 values. This list is compared directly to those in Table 1 to determine their overlapping fraction, which indicates the ability of our ML models to identify most-favorable systems. Using this metric, we can see that the accuracy of the prediction of these ML models as a function of training points used. In Fig. 13a, we can see that the overlapping percentage rapidly increases until around 1000 training data points, at which point the overlapping fraction values are maintained at ~65%. Meanwhile, if we

increase the intersection to 30%, we can see that the overlapping fraction can surpass 80% with as few as N=500 data points and reach as high as 90.9% when N increases. These results reassert the feasibility of the combinatorial approach described earlier, where users can use MD to generate small sets of data necessary to build a minimally functional *baseline* ML model to screen over a wider set of potential candidates. The results from such *baseline* models, can be utilized for various efforts, such as focusing on only simulating systems with the most favorable properties as predicted from the model or building more varied and evenly sampled data sets based on the predictive deficiencies from the baseline model to further improve its robustness. Either approach can improve the quality of results obtained with finite computing resources. Depending on the specific application, complexity of the systems of interest, and computing power, one can choose an optimal strategy to employ, *e.g.,* how many data should be collected to train ML models, and how much of the dataspace to be truncated based on suggestions by the models. However, it is worth recalling that the performance of the random forest regressor algorithm is dependent on the distribution of properties in the training data, and identifying *a priori* which systems to simulation to ensure appropriate distribution may be challenging. Solving this issue may require additional iterations of training ML models, where we create multiple predictive models as simulation data become available, using these ML models to suggest additional systems to ensure appropriate sampling, rather than identifying favorable candidates at this stage. Although such intermediate ML models, may not have high accuracy, they can provide valuable information needed to improve the performance of the subsequent ML models.

In Table 2 and 3 we summarize the outcome, R^2 and MAPE, of applying all of the ML models, *i.e.,* Summers *et al., 5050-train,* and *total-train* models, on all available test sets, *i.e., 5050-test, 2575-test,* and *total-test;* summary of other common metrics, *i.e.,* mean absolute error

(MAE) and mean squared error (MSE) is also included in the Supplemental Section (see Table S2 and S3).[65] Table 2 directly compares of the performance of the different ML models. We note, of all models trained with 100 data points, the performance of the *total-train* models exhibits the highest accuracy, followed by the *5050-train,* and finally models trained by Summers *et al.* data. The difference in performance likely results from the inclusion of mixed-monolayer systems, resulting in better distributed training sets provided by *total-train.* However, this trend does not persist for models trained with more data. Interestingly, we note that the models trained on the *5050-train* data appear to have better performance compared to the *total-train* models when predicting the *5050-test* set, since they require less data to acquire similar predictive ability. Focusing on the performance of the different models on the *total-test* set, which is expected to be more difficult to predict, the *5050-train* models show comparable accuracy to the *total-train* models. These observations trends are also confirmed by the MAPE values in Table 3. This result suggests that as long as the models are trained on a sufficiently large data set that captures the distribution of the population well, the models are extensible to untested regimes. That is, for significant variations, in our case composition, it may be more computationally efficient and accurate to train models with different compositions separately, rather than aggregating data into a large training set. Furthermore, all models trained on a relatively small amount of data, *e.g.,* 1000 data points, exhibit similar performance across all test sets. This reaffirms the transferability of these ML models from a more general perspective and highlights the feasibility of utilizing ML algorithms to estimate properties of systems whose designs may be dissimilar to the training data used. Expectedly, the ML models trained with 7816 data points from the *total-train* data set (80% of the total data set) demonstrate the best performance across all test sets, but of course, require the highest computational in terms of gathering training data.

Table 2. Summary of the performance of each ML models when predicting COF and F_0 for all different test sets, measured by R^2. For each data point, the R^2 value is averaged from R^2 of individual ML models (5 replicates) when applied to the test set.

	N	*5050-test*		*2575-test*		*Total-test*	
		COF	***F₀***	**COF**	***F₀***	**COF**	***F₀***
Summers *et al.* models	100	0.472	0.657	0.429	0.643	0.443	0.648
***5050-train* models**	100	0.543 ± 0.063	0.722 ± 0.058	0.475 ± 0.041	0.692 ± 0.031	0.496 ± 0.046	0.701 ± 0.032
	200	0.647 ± 0.012	0.792 ± 0.033	0.557 ± 0.008	0.742 ± 0.024	0.584 ± 0.008	0.757 ± 0.026
	300	0.687 ± 0.017	0.826 ± 0.011	0.584 ± 0.011	0.774 ± 0.007	0.615 ± 0.013	0.79 ± 0.007
	500	0.734 ± 0.004	0.859 ± 0.008	0.618 ± 0.005	0.819 ± 0.002	0.653 ± 0.004	0.831 ± 0.004
	1000	0.799 ± 0.007	0.885 ± 0.01	0.647 ± 0.001	0.848 ± 0.007	0.693 ± 0.002	0.859 ± 0.005
	1500	0.835 ± 0.011	0.907 ± 0.007	0.652 ± 0.004	0.867 ± 0.006	0.706 ± 0.005	0.879 ± 0.003
	2000	0.88 ± 0.004	0.92 ± 0.002	0.656 ± 0.007	0.875 ± 0.004	0.723 ± 0.006	0.888 ± 0.003
	2500	0.915 ± 0.003	0.927 ± 0.001	0.664 ± 0.003	0.881 ± 0.001	0.739 ± 0.002	0.895 ± 0.001
	2680	0.926 ± 0.0	0.928 ± 0.0	0.668 ± 0.0	0.882 ± 0.0	0.745 ± 0.0	0.896 ± 0.0
***Total-train* models**	100	0.567 ± 0.019	0.701 ± 0.027	0.511 ± 0.019	0.684 ± 0.023	0.528 ± 0.018	0.69 ± 0.019
	200	0.614 ± 0.032	0.792 ± 0.008	0.555 ± 0.02	0.763 ± 0.02	0.573 ± 0.021	0.772 ± 0.015
	300	0.652 ± 0.021	0.818 ± 0.017	0.595 ± 0.017	0.817 ± 0.014	0.612 ± 0.015	0.817 ± 0.014
	500	0.693 ± 0.02	0.849 ± 0.008	0.631 ± 0.011	0.855 ± 0.011	0.65 ± 0.01	0.853 ± 0.008
	1000	0.738 ± 0.007	0.883 ± 0.007	0.684 ± 0.007	0.89 ± 0.006	0.701 ± 0.005	0.888 ± 0.005
	1500	0.768 ± 0.006	0.897 ± 0.007	0.715 ± 0.005	0.904 ± 0.003	0.731 ± 0.004	0.902 ± 0.004
	2000	0.784 ± 0.006	0.908 ± 0.005	0.736 ± 0.003	0.911 ± 0.006	0.751 ± 0.003	0.91 ± 0.006
	2500	0.807 ± 0.006	0.915 ± 0.009	0.756 ± 0.004	0.918 ± 0.006	0.771 ± 0.003	0.917 ± 0.007
	4000	0.851 ± 0.006	0.928 ± 0.006	0.799 ± 0.004	0.928 ± 0.004	0.815 ± 0.003	0.928 ± 0.004
	6000	0.894 ± 0.003	0.937 ± 0.003	0.853 ± 0.005	0.936 ± 0.001	0.865 ± 0.004	0.936 ± 0.001
	7816	0.925 ± 0.0	0.943 ± 0.0	0.898 ± 0.0	0.941 ± 0.0	0.906 ± 0.0	0.942 ± 0.0

$^{\lambda}R^2$ when predicting coefficient of friction
$^{9}R^2$ when predicting adhesion force

Table 3. Summary of the performance of each ML models when predicting COF and F_0 for all different test sets, measured by MAPE. For each data point, the MAPE value is averaged from MAPE of individual ML models (5 replicates) when applied to the test set.

	N	*5050-test*		*2575-test*		*Total-test*	
		COF	***F₀***	**COF**	***F₀***	**COF**	***F₀***
Summers *et al.* models	100	0.0565	0.266	0.0653	0.341	0.0623	0.315
***5050-train* models**	100	0.0546 ± 0.003	0.259 ± 0.016	0.0643 ± 0.002	0.343 ± 0.018	0.061 ± 0.003	0.315 ± 0.016
	200	0.0478 ± 0.001	0.224 ± 0.011	0.0589 ± 0.001	0.312 ± 0.009	0.0551 ± 0.001	0.282 ± 0.01
	300	0.0447 ± 0.001	0.212 ± 0.007	0.0568 ± 0.001	0.295 ± 0.011	0.0527 ± 0.001	0.267 ± 0.009
	500	0.0407 ± 0.001	0.193 ± 0.004	0.0542 ± 0.0	0.275 ± 0.005	0.0496 ± 0.0	0.247 ± 0.004
	1000	0.0339 ± 0.001	0.175 ± 0.003	0.0521 ± 0.0	0.26 ± 0.003	0.0459 ± 0.0	0.231 ± 0.002
	1500	0.0301 ± 0.001	0.163 ± 0.001	0.052 ± 0.0	0.253 ± 0.004	0.0445 ± 0.001	0.222 ± 0.002
	2000	0.0251 ± 0.0	0.155 ± 0.002	0.0513 ± 0.001	0.248 ± 0.002	0.0423 ± 0.0	0.217 ± 0.001
	2500	0.0209 ± 0.0	0.152 ± 0.002	0.0508 ± 0.0	0.246 ± 0.001	0.0406 ± 0.0	0.213 ± 0.001
	2680	0.0196 ± 0.0	0.15 ± 0.0	0.0505 ± 0.0	0.245 ± 0.0	0.0399 ± 0.0	0.213 ± 0.0
***Total-train* models**	100	0.0539 ± 0.0012	0.27 ± 0.022	0.0621 ± 0.001	0.352 ± 0.04	0.0593 ± 0.0008	0.324 ± 0.033
	200	0.0504 ± 0.0025	0.228 ± 0.011	0.0592 ± 0.0016	0.304 ± 0.011	0.0562 ± 0.0016	0.278 ± 0.011
	300	0.0478 ± 0.0018	0.214 ± 0.005	0.0562 ± 0.0013	0.277 ± 0.007	0.0533 ± 0.0011	0.256 ± 0.006
	500	0.0447 ± 0.0017	0.197 ± 0.006	0.0533 ± 0.0011	0.262 ± 0.006	0.0504 ± 0.0009	0.24 ± 0.006
	1000	0.0408 ± 0.0005	0.178 ± 0.002	0.0486 ± 0.0008	0.24 ± 0.005	0.046 ± 0.0004	0.219 ± 0.003
	1500	0.0379 ± 0.0007	0.167 ± 0.003	0.0456 ± 0.0005	0.229 ± 0.005	0.043 ± 0.0004	0.208 ± 0.005
	2000	0.0359 ± 0.0009	0.162 ± 0.003	0.0433 ± 0.0004	0.225 ± 0.006	0.0407 ± 0.0004	0.204 ± 0.005
	2500	0.0336 ± 0.0005	0.157 ± 0.004	0.0411 ± 0.0004	0.22 ± 0.005	0.0385 ± 0.0002	0.198 ± 0.005
	4000	0.0287 ± 0.0006	0.148 ± 0.002	0.0361 ± 0.0006	0.21 ± 0.004	0.0336 ± 0.0003	0.189 ± 0.003
	6000	0.0236 ± 0.0004	0.145 ± 0.002	0.03 ± 0.0005	0.204 ± 0.002	0.0278 ± 0.0004	0.184 ± 0.002
	7816	0.0198 ± 0.0	0.14 ± 0.0	0.0246 ± 0.0001	0.2 ± 0.0	0.023 ± 0.0	0.179 ± 0.0

[λ] MAPE when predicting coefficient of friction
[9] MAPE when predicting adhesion force

As a proof-of-concept of using ML to pre-screen the design space, we perform a screening study using one of the *total-train* models (*model-0)* for COF and F_0 (trained with 7816 data points each). The chemical space for this screening was constructed by querying the ChEMBL small molecules library,[36,37] and identifying chemistries whose molecular weight ranges from 4 to 99

amu; this list of chemistries (981) underwent further filtering to remove those containing metallic elements and those that cannot be processed by the RDKit library, *e.g.,* chiral or charged molecules, resulting in 621 unique terminal group chemistries (provided in the Supplemental Repository[39]). With these 621 chemistries, 193,131 unique systems can be created in which each monolayer is homogeneous (*i.e.,* containing only one type of terminal group); mixed monolayer chemistries were not considered at the moment due to the vast amount of data that would be generated, a dynamically pruning approach to help drive the screening towards a smaller subset of the mixed monolayer systems is necessary to explore this space in any feasible time and memory requirements. This simple system design (dual homogeneous monolayers) was chosen to allow more unique chemistries to be considered in a reasonable time frame, since introduction of mixed monolayers would scale up the number of systems to be considered by several orders of magnitude. Descriptors for the 621 terminal groups were determined using the SMILES strings for each chemistry (as described in the Method section); these descriptors were then provided as input to the ML models which in turn predicted tribological properties for the 193,131 unique systems. This screening process, which evaluated 385,641 systems since duplicate systems (*i.e.,* systems in which chemistry A was the top monolayer and chemistry B on the bottom monolayer and vice versa) were not removed, took approximately 24 hours to predict the COF and F_0 values on a standard desktop computer (~ 0.22 seconds per system), which is orders of magnitudes faster than the time required to perform a single MD simulation and without the need for expansive computational resources.

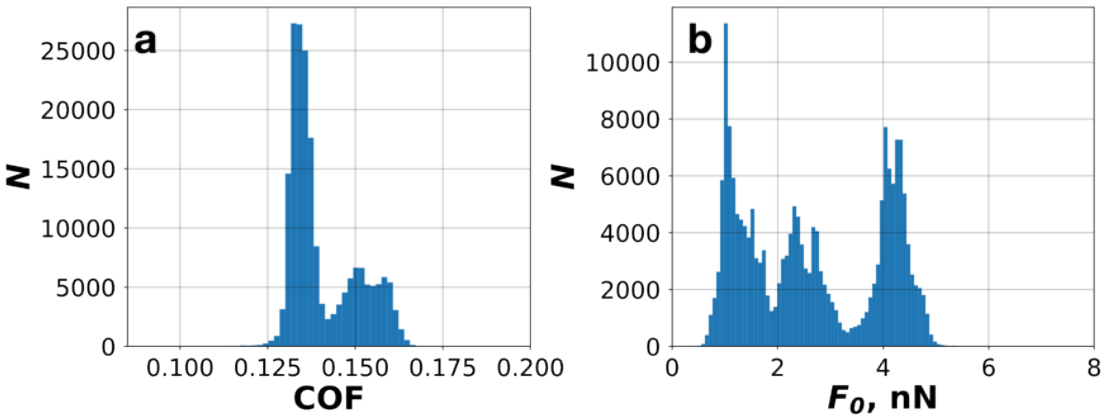

Figure 14. Distribution of (a) COF and (b) F_0 predicted by the ML models for 193,131 unique systems created with molecules from ChEMBL small molecules library.

Table 4. 20 best performing systems determined by the intersection of the top 2000 systems ranked by their COF and the top 2000 systems ranked by their F_0. The properties were predicted using one of the *total-train* models (*model-0)*.

	Terminal Group A	Terminal Group B	COF	F_0, nN
1	cyano	propyl	0.1144	0.7257
2	cyano	cyclopropyl	0.1151	0.4631
3	methyl	cyano	0.1153	0.5532
4	acetylene	1,1-difluoroethyl	0.118	0.7699
5	cyano	ethyl	0.1206	0.649
6	fulminic acid	cyclopropyl	0.1236	0.7117
7	ethylene	1,1-difluoroethyl	0.1244	0.7341
8	bromoethyl	1,2-diformylhydrazine	0.125	0.7695
9	methyl	fulminic acid	0.126	0.7704
10	cyano	difluoroethyl	0.1265	0.7279
11	bromoethyl	malononitrile	0.1269	0.7098
12	acetylene	ethyl	0.127	0.7254
13	1,1-difluoroethane	propene	0.1271	0.729
14	propyl	2,2-difluoroacetamide	0.128	0.7423
15	acetylene	propyl	0.1281	0.7777
16	methyl	acetylene	0.1281	0.7405
17	bromoethyl	1,2-dicyanoethyl	0.1282	0.7737
18	fulminic acid	ethyl	0.1283	0.7725
19	cyclopropyl	acrylonitrile	0.1283	0.7152
20	allyl	but-2-yne	0.1283	0.7546

The distribution of COF and F_0 of the systems predicted by the ML model are shown in Figure 13. We note the distributions differ from that of the data set screened using MD simulations (see Figs. 3 and 7), which is expected given the vastly expanded chemical design space. Using the first quartile of the COF distribution (0.1280) and F_0 distribution (0.8966 nN) obtained from MD as a reference, the data set contains 5121 systems that can be considered to have good COF values and 10,598 systems with good F_0 values. To further reduce the number of systems of interest, a list of 2000 best performing systems ranked by their COF values and a list of the 2000 best systems ranked by their F_0 are compiled, with the top 20 systems at the intersection of these lists reported in Table 3. We note that many of the same chemistries that were identified by our initial MD simulations (see Table 1) are also observed in this list; specifically, Systems 2, 3, and 16 have been considered in Summer *et al.*[8]*;* along with several other chemistries that may be worth future consideration, such as various alkenes (allyl, propene), alkynes (acetylene, but-2-yne), halocarbons (1,1-difluoroethyl, bromoethyl, vinyl chloride), and nitriles (cyano, malononitrile, acrylonitrile). We note that none of the systems reported in Table 3 outperform those previously identified in Table 1 from the MD simulations (in terms of COF and F_0), although this might be expected since the systems in Table 3 consist of 2 homogeneous monolayers, and hence, do not include the benefits offered by the mixed monolayers, as we discussed earlier. Nonetheless, this highlights the feasibility of combining ML with MD screening to reduce computational cost and identify favorable candidates for further study, in particular for reducing the vast design space of mixed monolayer systems using such a database for screening.

Conclusion

Utilizing the MoSDeF software suite with the Signac Framework[13,14], a workflow to initialize, parametrize, convert to MD engine inputs, perform MD simulations, calculate the tribological properties of interest, and then train a predictive ML model for these soft matter monolayer systems was developed and extended from our previous smaller scale study. The tribological properties of nearly 10,000 unique system designs have been screened. From the MD simulations, we identified systems that exhibited both low COF and F_0. The bulk of these systems consisted of a cyano group monolayer contacting a heterogeneous monolayer, suggesting mixed monolayer systems may provide a viable route for further tuning tribological performance. However, we note that no clear trends were observed for different mixing ratios in the monolayer, suggesting that their effects strongly depend on the chemistries involved. Although using high-throughput screening we were able to much more readily determine systems with favorable properties than could be accomplished through experiment, the process still requires a significant amount of time and computing resources. However, coupling MD simulations with machine learning can guide the screening process and reduce the simulations needed in order to optimize system designs. Using this approach, we have assessed the dependence of the based on the training set provided. The results suggest a positive correlation between the performance of machine learning models with the size of the training set, along with factors such as the distribution of the data; however, performance typically plateaus once a modest data set size is reached. For the type of systems considered in this study, a training set of 1000 data points is found to be sufficient to train an efficient predictive model. We note that at small training set sizes, *i.e.,* fewer than 500 data points, the machine learning models were still quite successful in determining the best and worst performing systems. Moreover, the models were shown to have high transferability when applied to predict properties of dissimilar systems and that improvements

in accuracy seen for larger training sets often do not necessarily equate to improved performance when models are transferred to systems outside of the training set. These findings suggest a synergistic approach of using MD simulations and machine learning to build high quality predictive models and minimize computing resources needed: MD can be used to generate a small set of data to train *baseline* ML models, which can then be utilized to quickly evaluate possible candidates and narrow the parameter space. The performance of the *baseline* model is dependent on the distribution of training data provided; however, the accuracy of the model can be improved via a few iterations of training, using earlier, less accurate, models to guide simulations toward creating well distributed training data. In addition, the *baseline* model could help confirm/provide insight about the connection between chemical intuition with properties of interest. We also note, that care must be taken to ensure that the data set is not overtly biased and that the further trained models are not overfit to the provided data. This work follows guidelines suggested by the TRUE standard, emphasizing the reproducibility and extensibility of the study; accordingly, the Supplementary Information contains all the information needed to reproduce the simulations and machine learning models described in this work. The code and data are distributed via GitHub.

Supplemental Information

See supplemental material for instructions to access the supplemental GitHub Repository containing data and analysis codes, additional forcefield details, and further discussion regarding the ML models utilized in this work.

Acknowledgement

Funding for this work is provided by the National Science Foundation (NSF) through Grants OAC-1835874. AMH also acknowledges support from the National Science Foundation through grant number DMR-1852157. This research used resources provided by the Office of Science of the Department of Energy at the Oak Ridge Leadership Computing Facility operated under Contract DE-AC05-00OR22725 via an award from the INCITE program and the National Energy Research Scientific Computing Center (NERSC) operated under Contract DE-AC02-05CH11231.

References

1 B. Bhushan and S. Sundararajan, Micro/nanoscale friction and wear mechanisms of thin films using atomic force and friction force microscopy, *Acta Mater.*, 1998, **46**, 3793 ï 3804.

2 N. S. Tambe and B. Bhushan, Nanotribological characterization of self-assembled monolayers deposited on silicon and aluminium substrates, *Nanotechnology*, 2005, **16**, 1549 ï1558.

3 S. G. Vilt, Z. Leng, B. D. Booth, C. McCabe and G. K. Jennings, Surface and frictional properties of two-component alkylsilane monolayers and hydroxyl-terminated monolayers on silicon, *J. Phys. Chem. C*, 2009, **113**, 14972 ï14977.

4 B. Yu, L. Qian, J. Yu and Z. Zhou, Effects of tail group and chain length on the tribological behaviors of self-assembled dual-layer films in atmosphere and in vacuum, *Tribol. Lett.*, 2009, **34**, 1 ï10.

5 N. J. Brewer, B. D. Beake and G. J. Leggett, Friction force microscopy of self-assembled monolayers: Influence of adsorbate alkyl chain length, terminal group chemistry, and scan velocity, *Langmuir*, 2001, **17**, 1970 ï1974.

6 T. Le, V. C. Epa, F. R. Burden and D. A. Winkler, Quantitative structure-property relationship modeling of diverse materials properties, *Chem. Rev.*, 2012, **112**, 2889 ï2919.

7 T. Bereau, Computational compound screening of biomolecules and soft materials by molecular simulations, *Model. Simul. Mater. Sci. Eng.*, 2021, **29**, 023001.

8 A. Z. Summers, J. B. Gilmer, C. R. Iacovella, P. T. Cummings and C. McCabe, MoSDeF, a Python Framework Enabling Large-Scale Computational Screening of Soft Matter: Application to Chemistry-Property Relationships in Lubricating Monolayer Films, *J.*

Chem. Theory Comput., 2020, **16**, 1779 ï1793.

9 J. L. Rivera, G. K. Jennings and C. McCabe, Examining the frictional forces between mixed hydrophobic ïhydrophilic alkylsilane monolayers, *J. Chem. Phys.*, 2012, **136**, 244701.

10 O. A. Mazyar, G. Kane Jennings and C. McCabe, Frictional dynamics of alkylsilane monolayers on SiO 2: Effect of 1-n-butyl-3-methylimidazolium nitrate as a lubricant, *Langmuir*, 2009, **25**, 5103 ï5110.

11 J. Ben Lewis, S. G. Vilt, J. L. Rivera, G. K. Jennings and C. McCabe, Frictional properties of mixed fluorocarbon/hydrocarbon silane monolayers: A simulation study, *Langmuir*, 2012, **28**, 14218 ï14226.

12 Molecular Simulation Design Framework (MoSDeF), https://mosdef.org.

13 V. Ramasubramani, C. Adorf, P. Dodd, B. Dice and S. Glotzer, signac: A Python framework for data and workflow management, *Proc. 17th Python Sci. Conf.*, 2018, 152 ï159.

14 C. S. Adorf, P. M. Dodd, V. Ramasubramani and S. C. Glotzer, Simple data and workflow management with the signac framework, *Comput. Mater. Sci.*, 2018, **146**, 220 ï229.

15 M. W. Thompson, R. Matsumoto, R. L. Sacci, N. C. Sanders and P. T. Cummings, Scalable Screening of Soft Matter: A Case Study of Mixtures of Ionic Liquids and Organic Solvents, *J. Phys. Chem. B*, 2019, **123**, 1340 ï1347.

16 M. W. Thompson, J. B. Gilmer, R. A. Matsumoto, C. D. Quach, P. Shamaprasad, A. H. Yang, C. R. Iacovella, C. McCabe and P. T. Cummings, Towards molecular simulations that are transparent, reproducible, usable by others, and extensible (TRUE), *Mol. Phys.*, 2020, **0**, e1742938.

17 P. T. Cummings, C. McCabe, C. R. Iacovella, A. Ledeczi, E. Jankowski, A. Jayaraman, J. C. Palmer, E. J. Maginn, S. C. Glotzer, J. A. Anderson, J. I. Siepmann, J. J. Potoff, R. A. Matsumoto, J. B. Gilmer, R. S. DeFever, R. Singh and B. Crawford, Open-Source Molecular Modeling Software in Chemical Engineering, with Focus on the Molecular Simulation Design Framework (MoSDeF), *AIChE J.*, 2020, to be published.

18 F. A. Faber, L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley and O. A. Von Lilienfeld, Prediction Errors of Molecular Machine Learning Models Lower than Hybrid DFT Error, *J. Chem. Theory Comput.*, 2017, **13**, 5255–5264.

19 J. S. Smith, O. Isayev and A. E. Roitberg, ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost, *Chem. Sci.*, 2017, **8**, 3192–3203.

20 J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, Highly accurate protein structure prediction with AlphaFold, *Nature*, 2021, **596**, 583–589.

21 S. Doerr, M. Majewski, A. Pérez, A. Krämer, C. Clementi, F. Noe, T. Giorgino and G. De Fabritiis, TorchMD: A Deep Learning Framework for Molecular Simulations, *J. Chem. Theory Comput.*, 2021, **17**, 2355–2363.

22 J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev and A. E. Roitberg, Less is more: Sampling chemical space with active learning, *J. Chem. Phys.*, 2018, **148**, 241733.

23 H. Tian, F. Trozzi, B. D. Zoltowski and P. Tao, Deciphering the Allosteric Process of the Phaeodactylum tricornutum Aureochrome 1a LOV Domain, *J. Phys. Chem. B*, 2020, **124**, 8960–8972.

24 F. Wang, L. Shen, H. Zhou, S. Wang, X. Wang and P. Tao, Machine Learning Classification Model for Functional Binding Modes of TEM-1 β-Lactamase, *Front. Mol. Biosci.*, 2019, **6**, 1–18.

25 K. Shmilovich, R. A. Mansbach, H. Sidky, O. E. Dunne, S. S. Panda, J. D. Tovar, A. L. Ferguson, E. Olivia, S. S. Panda, J. D. Tovar and A. L. Ferguson, Discovery of Self-Assembling π-Conjugated Peptides by Active Learning-Directed Coarse-Grained Molecular Simulation, *J. Phys. Chem. B*, 2020, **124**, 3873–3891.

26 A. S. Kelkar, B. C. Dallin and R. C. Van Lehn, Identifying nonadditive contributions to the hydrophobicity of chemically heterogeneous surfaces via dual-loop active learning, *J. Chem. Phys.*, 2022, **156**, 024701.

27 J. P. Janet, S. Ramesh, C. Duan and H. J. Kulik, Accurate Multiobjective Design in a Space of Millions of Transition Metal Complexes with Neural-Network-Driven Efficient Global Optimization, *ACS Cent. Sci.*, 2020, **6**, 513–524.

28 R. J. Gowers, A. H. Farmahini, D. Friedrich and L. Sarkisov, Automated analysis and benchmarking of GCMC simulation programs in application to gas adsorption, *Mol. Simul.*, 2018, **44**, 309–321.

29 K. M. Jablonka, G. M. Jothiappan, S. Wang, B. Smit and B. Yoo, Bias free multiobjective active learning for materials design and discovery, *Nat. Commun.*, 2021, **12**, 2312.

30 M. A. F. Afzal, M. Haghighatlari, S. P. Ganesh, C. Cheng and J. Hachmann, Accelerated Discovery of High-Refractive-Index Polyimides via First-Principles Molecular Modeling,

Virtual High-Throughput Screening, and Data Mining, *J. Phys. Chem. C*, 2019, **123**, 14610 ï14618.

31 C. Kuenneth, W. Schertzer and R. Ramprasad, Copolymer Informatics with Multitask Deep Neural Networks, *Macromolecules*, 2021, **54**, 5957 ï5961.

32 C. Kim, R. Batra, L. Chen, H. Tran and R. Ramprasad, Polymer design using genetic algorithm and machine learning, *Comput. Mater. Sci.*, 2021, **186**, 110067.

33 A. J. Gormley and M. A. Webb, Machine learning in combinatorial polymer chemistry, *Nat. Rev. Mater.*, 2021, **6**, 642 ï644.

34 A. Statt, D. C. Kleeblatt and W. F. Reinhart, Unsupervised learning of sequence-specific aggregation behavior for a model copolymer, *Soft Matter*, 2021, **17**, 7697 ï7707.

35 M. A. Webb, N. E. Jackson, P. S. Gil and J. J. de Pablo, Targeted sequence design within the coarse-grained polymer genome, *Sci. Adv.*, 2020, **6**, eabc6216.

36 A. Gaulton, A. Hersey, M. Nowotka, A. P. Bento, J. Chambers, D. Mendez, P. Mutowo, F. Atkinson, L. J. Bellis, E. Cibrián-Uhalte, M. Davies, N. Dedman, A. Karlsson, M. P. Magariños, J. P. Overington, G. Papadatos, I. Smit and A. R. Leach, The ChEMBL database in 2017, *Nucleic Acids Res.*, 2017, **45**, D945 ïD954.

37 M. Davies, M. Nowotka, G. Papadatos, N. Dedman, A. Gaulton, F. Atkinson, L. Bellis and J. P. Overington, ChEMBL web services: streamlining access to drug discovery data and utilities, *Nucleic Acids Res.*, 2015, **43**, W612 ïW620.

38 A. Z. Summers, C. R. Iacovella, P. T. Cummings, and C. McCabe, Investigating Alkylsilane Monolayer Tribology at a Single-Asperity Contact with Molecular Dynamics Simulation, *Langmuir*, 2017, **33**, 11270 ï11280.

39 C. D. Quach, J. B. Gilmer, D. Pert, A. Mason-Hogans, C. R. Iacovella, P. T. Cummings

and C. McCabe, High-Throughput Screening of Tribological Properties of Monolayer Films using Molecular Dynamics and Machine Learning: Supplemental Repository, https://github.com/daico007/iMoDELS-supplements/, (accessed 10 June 2021).

40 J. E. Black, C. R. Iacovella, P. T. Cummings and C. McCabe, Molecular Dynamics Study of Alkylsilane Monolayers on Realistic Amorphous Silica Surfaces, *Langmuir*, 2015, **31**, 3086–3093.

41 M. Chandross, G. S. Grest and M. J. Stevens, Friction between alkylsilane monolayers: Molecular simulation of ordered monolayers, *Langmuir*, 2002, **18**, 8392–8399.

42 P. T. Mikulski and J. A. Harrison, Packing-density effects on the friction of n-alkane monolayers, *J. Am. Chem. Soc.*, 2001, **123**, 6873–6881.

43 K. Kojio, S. Ge, A. Takahara and T. Kajiyama, n-Octadecyltrichlorosilane Monolayer Prepared at an Air / Water Interface, *Society*, 1998, **14**, 1996–1999.

44 mBuild Github Repository, https://github.com/mosdef-hub/mbuild, (accessed 17 August 2018).

45 Foyer Github Repository, https://github.com/mosdef-hub/foyer, (accessed 10 August 2020).

46 C. Klein, J. Sallai, T. J. Jones, C. R. Iacovella, C. McCabe, and P. T. Cummings, in *Foundations of Molecular Modeling and Simulation. Molecular Modeling and Simulation (Applications and Perspectives)*, eds. R. Q. Snurr, C. S. Adjiman and D. A. Kofke, Springer, Singapore, Singapore, 2016, pp. 79–92.

47 C. Klein, A. Z. Summers, M. W. Thompson, J. B. Gilmer, C. McCabe, P. T. Cummings, J. Sallai and C. R. Iacovella, Formalizing atom-typing and the dissemination of force fields with foyer, *Comput. Mater. Sci.*, 2019, **167**, 215–227.

48 W. L. Jorgensen, D. S. Maxwell and J. Tirado-Rives, Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids, *J. Am. Chem. Soc.*, 1996, **118**, 11225 ï11236.

49 H. J. C. Berendsen, D. van der Spoel and R. van Drunen, GROMACS: A message-passing parallel molecular dynamics implementation, *Comput. Phys. Commun.*, 1995, **91**, 43 ï56.

50 M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess and E. Lindah, Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers, *SoftwareX*, 2015, **1 ï2**, 19 ï25.

51 C. D. Lorenz, E. B. Webb, M. J. Stevens, M. Chandross and G. S. Grest, Frictional dynamics of perfluorinated self-assembled monolayers on amorphous SiO2, *Tribol. Lett.*, 2005, **19**, 93 ï98.

52 Signac Framework, https://signac.io/.

53 S. Plimpton, Fast Parallel Algorithms for Short-Range Molecular Dynamics, *J. Comput. Phys.*, 1995, **117**, 1 ï19.

54 A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in ô t Veld, A. Kohlmeyer, S Stevens, J. Tranchida, C. Trott and S. J. Plimpton, LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, *Comput. Phys. Commun.*, 2022, **271**, 108171.

55 W. G. Hoover, Canonical dynamics: Equilibrium phase-space distributions, *Phys. Rev. A*, 1985, **31**, 1695 ï1697.

56 T. Darden, D. York and L. Pedersen, Particle mesh Ewald: An N ïlog(N) method for Ewald sums in large systems, *J. Chem. Phys.*, 1993, **98**, 10089 ï10092.

57 S. C. Clear and P. F. Nealey, Chemical force microscopy study of adhesion and friction between surfaces functionalized with self-assembled monolayers and immersed in solvents, *J. Colloid Interface Sci.*, 1999, **213**, 238 ï250.

58 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and É. Duchesnay, Scikit-learn: Machine Learning in Python, *J. Mach. Learn. Res.*, 2011, **12**, 2825 ï2830.

59 Y. L. Pavlov, *Random Forests* , De Gruyter, First. | Boca Ra

60 V. Svetnik, A. Liaw, C. Tong, J. Christopher Culberson, R. P. Sheridan and B. P. Feuston, Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling, *J. Chem. Inf. Comput. Sci.*, 2003, **43**, 1947 ï1958.

61 RDKit: Open-source cheminformatics, http://rdkit.org/.

62 D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules, *J. Chem. Inf. Model.*, 1988, **28**, 31 ï36.

63 R. A. Patel, C. H. Borca and M. A. Webb, Featurization Strategies for Polymer Sequence or Composition Design by Machine Learning, 2021, 1 ï25.

64 A. Z. Summers, atools, https://github.com/PTC-CMC/atools.git.

65 G. Vishwakarma, A. Sonpal and J. Hachmann, Metrics for Benchmarking and Uncertainty Quantification: Quality, Applicability, and Best Practices for Machine Learning in Chemistry, *Trends Chem.*, 2021, **3**, 146 ï156.